

Aalto-yliopisto
Perustieteiden korkeakoulu
Tietotekniikan koulutusohjelma

Jaakko Kaisanlahti

Älylaitteiden kytkeminen terveydenhuollon tietojärjestelmiin

Diplomityö
Espoo, 21. syyskuuta 2016

Valvoja: Professori Kary Främling
Ohjaaja: Kauppatieteiden maisteri Jorma Kaista

Aalto University
 School of Science
 Degree Programme in Computer Science and Engineering

ABSTRACT OF
 MASTER'S THESIS

Author:	Jaakko Kaisanlahti		
Title:	Connecting smart devices to healthcare information systems		
Date:	September 21, 2016	Pages:	57
Major:	Software Technology	Code:	T-106
Supervisor:	Professor Kary Främling		
Advisor:	Jorma Kaista M.Sc. (Econ.)		
<p>The Internet of Things can be utilized in healthcare to treat and to prevent disease. However, public healthcare might only have enough resources to offer suitable devices to those who require immediate treatment. Utilizing devices that consumers own would allow healthcare personnel to prevent disease in healthy individuals.</p> <p>A system that connects consumer level smart devices to a healthcare information system was developed in this thesis. A group of simulated devices offers data in the O-DF format (Open Data Format). The data is sent over the O-MI (Open Messaging Interface) protocol to a transformer that translates it to the FHIR (Fast Healthcare Interoperability Resources) standard, which is used in healthcare information systems.</p> <p>The main problems discovered during the implementation were choosing how to represent health data in the O-DF format and how to map O-DF fields to their corresponding FHIR fields. To implement the system for production use, the relevant standards would have to become more widespread or middleware to translate device specific data to the O-DF format would have to be developed.</p>			
Keywords:	Internet of things, IoT, Open Data Format, O-DF, Open Mes- saging Interface, O-MI, Fast Healthcare Interoperability Re- sources, FHIR, healthcare		
Language:	Finnish		

Aalto-yliopisto
 Perustieteiden korkeakoulu
 Tietotekniikan koulutusohjelma

DIPLOMITYÖN
 TIIVISTELMÄ

Tekijä:	Jaakko Kaisanlahti		
Työn nimi:	Älylaitteiden kytkeminen terveydenhuollon tietojärjestelmiin		
Päiväys:	21. syyskuuta 2016	Sivumäärä:	57
Pääaine:	Ohjelmistotekniikka	Koodi:	T-106
Valvoja:	Professori Kary Främling		
Ohjaaja:	Kauppatieteiden maisteri Jorma Kaista		
<p>Esineiden internetiä voidaan hyödyntää terveydenhuollossa sairauksien hoitoon ja ennaltaehkäisyyn. Julkisella terveydenhuollolla ei kuitenkaan välttämättä ole resursseja tarjota siihen soveltuvia laitteita muille kuin hoitoa tarvitseville. Kuluttajilla olevien laitteiden hyödyntäminen mahdollistaisi perusterveiden ihmisten terveyden seurannan ja sairauksien ennaltaehkäisyn.</p> <p>Tässä työssä kehitettiin tapa yhdistää kuluttajatasen älylaitteita terveydenhuollon tietojärjestelmiin. Joukko simuloituja laitteita tarjosi dataa O-DF-muodossa (Open Data Format). Data siirrettiin O-MI-protokollalla (Open Messaging Interface) muuntimelle, joka käänsi sen terveydenhuollossa käytettävän FHIR-standardin (Fast Healthcare Interoperability Resources) mukaiseen muotoon.</p> <p>Suurimmat havaitut ongelmat liittyivät O-DF-datan esitysmuodon valintaan ja datakenttien vastaavuuksien tunnistamiseen O-DF ja FHIR-muodoissa. Järjestelmän toteuttaminen tuotantokäyttöön vaatisi käytettyjen standardien yleistymistä tai laitekohtaisten datamuunnoksen tekevien väliohjelmistojen kehitystä.</p>			
Asiasanat:	Esineiden internet, Open Data Format, O-DF, Open Messaging Interface, O-MI, Fast Healthcare Interoperability Resources, FHIR, terveydenhuolto		
Kieli:	Suomi		

Kiitossanat

Ensiksi haluan kiittää valvojaani Kary Främlingiä ja ohjaajaani Jorma Kais-
taa palautteesta, ideoista ja tsemppauksesta. Varsinkin Jorman kanssa pide-
tyt viikottaiset palaverit auttoivat rytmittämään työtä.

Olen kiitollinen työnantajalleni Accenturelle siitä, että sain käyttää työaika-
diplomityön tekemiseen. Erityisesti kiitän projektin väkeä siitä, että sain olla
pois ruodusta kahtena päivänä viikossa.

Kunniamaininta ERPG:n porukalle. Yhteispuhina on ollut tosi kivaa.

Lopuksi kiitän länsimetroa valmistumisen kirittämisestä. Olinpas nopeampi.

Espoo, 21. syyskuuta 2016

Jaakko Kaisanlahti

Sisältö

1	Johdanto	8
1.1	Tutkimuskysymykset	9
1.2	Työn rakenne	9
2	Tausta	10
2.1	Esineiden internet	10
2.2	Esineiden internet terveydenhuollossa	11
2.3	Valitut standardit	13
2.3.1	Standardi älylaitteille	13
2.3.2	Standardi terveydenhuollon tietojärjestelmille	15
3	Open Data Format ja Open Messaging Interface -standardit	17
3.1	Open Data Format	17
3.2	Open Messaging Interface	19
3.2.1	Kertaluku	20
3.2.2	Tilausviesti	20
3.2.3	Peruutusviesti	22
3.2.4	Kirjoitusviesti	22
4	Fast Healthcare Interoperability Resources	23
4.1	Standardin tila	23
4.2	Resurssit	24
4.2.1	Potilasresurssi	25

4.2.2	Havaintoresurssi	26
4.2.3	Laiteresurssi	27
4.3	RESTful-rajapinta	28
5	Toteutus	29
5.1	Skenaario	29
5.2	Toteutusvaihtoehdot	30
5.2.1	Puhdas FHIR-järjestelmä	30
5.2.2	Puhdas O-DF/O-MI-järjestelmä	31
5.2.3	Sekajärjestelmä	32
5.3	Järjestelmän osat	33
5.3.1	Äylaitteet	34
5.3.1.1	Vaaka	35
5.3.1.2	Älyranneke	36
5.3.1.3	Verenpainemittari	37
5.3.2	O-MI-palvelin	37
5.3.3	O-DF–FHIR-muunnin	38
5.3.4	FHIR-palvelin	39
5.4	Rajaukset	40
5.5	Lähdekoodi	40
6	Tulokset	41
6.1	O-DF-datan esitysmuodon valinta	41
6.2	Kenttien tunnistaminen	44
6.3	Laitteiden rekisteröinti	45
6.4	Mittausten yksiköt	46
6.5	FHIR-resurssien linkittäminen	46
7	Pohdinta	49
7.1	Sovellettavuus	49
7.2	Jatkokehitys	50

Luku 1

Johdanto

Internet kasvaa nopeasti. Kansainvälisen televiestintäliiton mukaan internetin käyttäjien määrä on kasvanut vuosina 2005–2015 2,2 miljardilla yhteensä 3,2 miljardiin käyttäjään [27]. Internetiin on kytketty laitteita niin paljon, että Internet Protocol Version 4:n (IPv4) 4,3 miljardin osoitteen osoiteavaruus on loppumassa kesken. [2]

Perinteisesti internetiä on käytetty päätelaitteilta, kuten tietokoneilta ja matkapuhelimilta. Tietokoneita on alettu upottaa muihin esineisiin, jotka on kytketty internetiin. Ilmiötä kutsutaan esineiden internetiksi (engl. Internet of Things, IoT). Esimerkiksi älypuhelimien kautta säädettävä asunnon termos- taatti on osa esineiden internetiä. Arviot internetiin kytkettyjen esineiden määrästä vaihtelevat: Tutkimusyhtiö Gartner ennustaa 6,4 miljardia laitetta vuodelle 2016 [12]. Juniper Research laskee, että niitä oli 13,4 miljardia vuonna 2015, ja arvioi määrän kasvavan vuoteen 2020 mennessä 38,5 miljardiin [26]. Verkkolaiteyritys Ciscon mukaan esineiden internet tuottaa lisäarvoa vuosina 2013–2022 14,4 biljoonaa Yhdysvaltain dollaria [4, s. 1].

Esineiden internetiä hyödynnetään terveydenhoidossa. Kuluttajille on tarjolla laitteita, jotka mittaavat terveystietoa käyttäjästä ja lataavat sen internetiin, josta käyttäjä voi sitä seurata. Älypuheliiniin on sisäänrakennettu terveysantureita, kuten askel- tai sykemittari. Näiden kuluttajille suunnattujen tuotteiden lisäksi on tarjolla lääketieteen ammattilaisten käyttöön tarkoitettuja laitteita. Näistä yksi esimerkki on implantti, joka annostelee osteoporoosilääkettä automaattisesti, ja jonka tilaa ja annostuksen aikataulua voi muuttaa langattomasti [9]. Cisco arvioi, että terveydenhuollon osa esineiden internetin tuottamasta lisäarvosta on 106 miljardia Yhdysvaltain dollaria [4, s. 13].

Älyesineiden valmistajat eivät ole sopineet keskenään esineiden internetin standardeista. Laitteilla on omat rajapinnat, ja ne on suunniteltu toimimaan vain valmistajan ekosysteemissä. Eri valmistajien laitteiden käyttäminen yhdessä ja ulkopuolisten järjestelmien kanssa on vaikeaa.

Samanaikaisesti terveydenhuollon tieto on siirtynyt sähköisiin järjestelmiin. Suomessa Apotti-hanke kehittää pääkaupunkiseudun terveydenhuollolle yhtenäistä tietojärjestelmää [3]. Uusien älylaitteiden pitää pystyä tallentamaan tietoa terveydenhuollon järjestelmiin automaattisesti.

Tämä diplomityö esittää yhden tavan yhdistää terveydenhuollon älyesineitä terveydenhuollon tietojärjestelmiin. Sekä älyesineille että tietojärjestelmälle on valittu standardi, ja niiden yhteensopivuutta on tutkittu.

1.1 Tutkimuskysymykset

Tämä diplomityö pyrkii vastaamaan kahteen tutkimuskysymykseen:

1. Kuinka terveydenhuollon älylaitteita voidaan yhdistää terveydenhuollon tietojärjestelmään?
2. Mitä haasteita yhdistämisessä on, ja kuinka ne voidaan ratkaista?

Kysymyksiin vastataan toteuttamalla yksinkertainen ohjelmistojärjestelmä, joka simuloi terveydenhuollon tietojärjestelmän kanssa kommunikoivia älylaitteita.

1.2 Työn rakenne

Luvussa 2 taustoitetaan esineiden internetiä, sen käyttöä terveydenhuollossa ja miten esineiden yhteensopivuus toteutetaan. Luvut 3 ja 4 selostavat valittuja laitteiden ja terveydenhuollon tietojärjestelmän standardeja. Luku 5 kuvaa, minkälainen järjestelmä toteutettiin, ja sitä seuraava luku 6 kertoo, mitä ongelmia toteutettaessa havaittiin ja miten ne ratkaistiin. Kappaleessa 7 mietitään, kuinka saatuja tuloksia voidaan soveltaa ja miten järjestelmää voisi jatkokehittää. Viimeinen luku 8 on päätäntö, jossa on työn yhteenveto.

Luku 2

Tausta

Tässä luvussa käydään läpi työn taustaa ja perustellaan, miksi työ on tehty. Ensimmäinen alaluku 2.1 määrittelee esineiden internetin käsitteen ja esittelee muutaman yleisen sovellusalueen. Sitä seuraava luku 2.2 kertoo lyhyesti, miten esineiden internetiä hyödynnetään terveydenhuollossa, ja esittää ongelman, johon tässä työssä pyritään löytämään ratkaisu. Viimeisessä alaluvussa 2.3 valitaan älylaitteille ja terveydenhoidon tietojärjestelmälle standardit, joita käyttäen järjestelmä toteutetaan.

2.1 Esineiden internet

Esineiden internetille ei ole virallista määritelmää. European Research Cluster on the Internet of Things määrittelee sen maailmanlaajuisena verkkona, jossa fyysisillä ja virtuaalisilla asioilla ja esineillä on identiteetti, fyysiset ominaisuudet ja virtuaalinen henkilöllisyys, ja ne ovat saumattomasti integroitu tietoverkkoon. Niiden odotetaan osallistuvan aktiivisesti liiketoiminnan ja tiedon prosesseihin. Ne voivat kommunikoida ja vuorovaikuttaa toistensa ja ympäristönsä kanssa. Ne keräävät tietoa ympäristöstään ja reagoivat sen muutoksiin ilman suoraa ihmisten väliintuloa. [39, s. 10]

Esineiden internet koostuu älykkäistä esineistä, eli esineistä, joihin on sisällytetty tietokone ja jotka on yhdistetty internetiin. Sen mahdollisuudet eivät perustu niinkään yhden älykkään esineen kykyihin, vaan niiden määrään: esineiden internetiin uskotaan yhdistettävän miljardeja laitteita. Lopputuloksena on älykäs planeetta, jossa jokapäiväiset ympäristön esineet ovat älykkäitä, mikä mahdollistaa resurssien tehokkaamman käytön. [28, s. 308–309]

Yksi esineiden internetin sovelluskohde on teollisuus. Teollinen internet on General Electricin lanseeraama termi, joka tarkoittaa esineiden internetin hyödyntämistä teollisuudessa. Sitä käytetään teolliseen automaatioon yhdistämällä sensoreita, hallintajärjestelmiä ja robotteja. GE ennustaa, että teollisella automaatiolla on samanlainen vaikutus maailmaan kuin teollisella valankumouksella. [35, s. 110]

Toinen jokapäiväistä elämää lähempänä oleva sovelluskohde on kotiautomaatio. Sisällyttämällä älykkäitä esineitä kotiin voidaan sen valaistusta, lämpötilaa ja muita ominaisuuksia hallita automaattisesti. Konkreettisin hyöty tästä on energiansäästö. Tarkan energianmittauksen on osoitettu vähentävän energiankulutusta 5–15 prosenttia [6, s. 17]. Eräs kodinomistaja oli laskenut sähkönkulutustaan kolmasosalla; Verrua Savoian kylä Italiassa oli leikannut sähkönkulutustaan 50 prosenttia ja vedenkulutusta 30 prosenttia. [32]

Tässä työssä keskitytään esineiden internetin sovelluksiin terveydenhuollossa, mitä käsitellään tarkemmin seuraavassa luvussa.

2.2 Esineiden internet terveydenhuollossa

Kuluttajille sunnattuja terveystureita on ollut saatavilla jo vuosia: esimerkiksi monilla lenkkeilijöillä on sykemittari. Tekniikan kehittyessä anturit on yhdistetty internetiin, ja niitä on lisätty muihin laitteisiin. Withings¹ valmistaa monia laitteita, jotka pystyvät lähettämään mittaustuloksensa käyttäjän tietojärjestelmään, esimerkiksi puhelinsovellukseen, josta niitä voi seurata. Moniin älypuhelimiin on lisätty askelmittari ja kalorilaskuri. Perinteisesti laitteet ovat toimineet vain valmistajan sovellusten kanssa, mutta nykyään niiden dataa voivat käyttää muutkin tahot. Esimerkiksi elektroniikkavalmistaja Apple tarjoaa laitteissaan HealthKit-ohjelmointirajapintaa², jolla kehittäjät pystyvät käyttämään laitteiden terveysominaisuuksia ja vuorovaikuttamaan muiden terveysohjelmien kanssa samalla laitteella.

Esineiden internetiä käytetään myös ammattimaisessa terveydenhuollossa. K. M. C. Kumar kirjoittaa ”kuntoiluesineiden internetistä”, jossa puettavat terveysturrit kommunikoivat verkkopalvelujen kanssa, joihin myös käyttäjää hoitavilla lääkäreillä on pääsy. Esimerkkinä käytettiin uniapnean hoitoa: Uniapneapotilailla on kohonnut riski kuolla äkilliseen sydänhalvaukseen yöllä. Seuraamalla potilasta älylaitteilla hänen nukkuessaan hänet voidaan herättää, jos elintoiminnoissa havaitaan vaarallisia poikkeavuuksia. Lisäksi

¹<http://www.withings.com>

²<https://developer.apple.com/healthkit/>

voidaan tehdä automaattinen hälytys, jos potilas saa sairauskohtauksen. [30] Toinen sovelluskohde on älylaitteiden käyttö hoitokodissa, jossa ne voivat hälyttää hoitohenkilökunnan automaattisesti ongelmatapauksissa [5]. Esineiden internetiä on myös käytetty välikyviin valojen aiheuttamien terveysriskien hallitsemiseen [7].

Esineiden internetin käytölle terveydenhuollossa on esitetty useita erilaisia arkkitehtuurimalleja. Ray esittää viisikerroksisen Home Health Hub Internet of Things -mallin vanhusten kotihoidolle. Sen kerrokset ovat:

- Fyysinen anturitaso: laitteet, jotka tekevät havaintoja.
- Paikallinen viestintätaso: 10–900 metrin etäisyyksillä tapahtuva viestintä antureista ylemmälle tasolle.
- Tietonkäsittelytaso: mikrokontrollerit, jotka käsittelevät tietoa ylemmien tasojen tarpeisiin.
- Internetsovellustaso: alemmilla tasoilla saatu tieto siirretään verkon yli puhelimille tai pilveen varastoitavaksi ja analysoitavaksi.
- Käyttäjäsovellustaso: terveydenhoitohenkilökunnan käyttämät sovellukset. [33]

Ullahin, Shahin ja Shanin k-Healthcare-arkkitehtuurinmallissa on neljä tasoa: anturit, verkko, internet ja palvelut [38]. Se on huomattavan samanlainen kuin Rayn malli: anturitaso kattaa Rayn kolme ensimmäistä tasoa, verkko- ja internettasot ovat Raylla yksi internettaso, ja palvelutaso vastaa käyttäjäsovellustasoa. Molemmissa malleissa voidaan erottaa kaksi eri tahoja: potilasta mittaavat laitteet ja terveydenhuollon käyttämät sovellukset ja tietojärjestelmät.

Älylaitteita voidaan käyttää ammattimaisen terveydenhuollon tukena kolmella tavalla: Ensimmäinen on käyttää niitä osana sairaalan infrastruktuuria, jossa sairaala hallitsee koko järjestelmää. Toinen on antaa tarvittavat laitteet potilaille kotikäyttöön. Kolmas tapa on hyödyntää potilailla jo olevia laitteita. Kahdessa ensimmäisessä tapauksessa sairaalan IT-ylläpito hallitsee kaikkia laitteita, ja voi taata niiden yhteensopivuuden. Kolmannessa tapauksessa käyttäjien omat laitteet eivät ole terveydenhuollon hallinnassa, eivätkä ne välttämättä toimi yhteen sen tietojärjestelmien kanssa.

Julkisen terveydenhuollon odotetaan tarjoavan kaikki hoitoon tarvittavat lääkkeet ja tarvikkeet ilmaiseksi tai hyvin edullisesti. Jos älyesineiden käyttö hoidossa yleistyy, julkinen terveydenhuolto luultavasti tarjoaa ne potilaille.

Sen sijaan julkisella terveydenhuollolla ei välttämättä ole resursseja tarjota perusterveille ihmisille laitteita automaattiseen terveydentilan seuraamiseen, vaikka siitä olisikin paljon hyötyä sairauksien ennaltaehkäisyssä. Tämän vuoksi olisikin tarpeellista kyetä yhdistämään mielivaltaisia kuluttajatasen terveyslaitteita terveydenhuollon järjestelmiin. Jotta tämä on mahdollista järkevin kustannuksin, kommunikointiprotokollat pitää jollain tasolla standardoida. Seuraavassa luvussa valitaan standardit toteuttavalle järjestelmälle.

2.3 Valitut standardit

Tässä työssä halutaan toteuttaa järjestelmä, joka pystyy yhdistämään mielivaltaisia tiettyä standardia toteuttavia älylaitteita terveydenhuollon tietojärjestelmään. Tätä varten tulee valita standardi tiedon esittämiseen ja siirtämiseen sekä esineiden internetin että tietojärjestelmien puolella.

Open Systems Interconnection Reference Model eli OSI-malli kuvaa tiedonsiirron protokollapinnon seitsemässä tasossa. Jokainen taso käsittelee dataa eri tavalla kuin muut. [1, s. 6] Kuvassa 2.1 on esitetty OSI-mallin tasot. Tässä työssä olemme kiinnostuneita päällimmäisen tason eli sovellustason protokollista, koska ne ovat vastuussa käyttäjille näkyvistä palveluista [1, s. 13–14].

7. Sovelluskerros
6. Esitystapakerros
5. Istuntokerros
4. Kuljetuskerros
3. Verkkokerros
2. Siirtokerros
1. Fyysinen kerros

Kuva 2.1: OSI-mallin tasot.

2.3.1 Standardi älylaitteille

Filippov on tehnyt vertailun esineiden internetiin soveltuvista protokollista. Vertaillut protokollat olivat Message Queue Telemetry Transport (MQTT),

Constrained Application Protocol (CoAP), Extensible Message and Presence Protocol (XMPP), Advanced Message Protocol (AMQP) ja Open Messaging Interface (O-MI). Niitä vertailtiin 14 kriteerillä:

1. Onko protokolla sovellusriippumaton?
2. Voiko sekä asiakas että palvelin aloittaa yhteyden?
3. Tukeeko tai tarvitseeko protokolla sitä, että viesti kulkee välittäjäta-
hojen kautta ennen kuin se saapuu määränpäähänsä?
4. Mikä on protokollan datan pysyvyys?
5. Onko protokollassa tilausominaisuus, eli voiko sillä pyytää tapahtumiin
tai aikaan perustuvia päivityksiä datasta?
6. Sisältävätkö protokollan viestit kaiken, mitä viestin ymmärtämiseen
tarvitaan?
7. Riippuuko protokolla jostain tietystä alemman tason protokollasta?
8. Tukeeko protokolla asynkronista ja synkronista kommunikointia?
9. Takaako protokolla viestien perillepääsyn?
10. Onko *piggy backing* tuettu, eli voiko vastausviesteihin sisällyttää uusia
pyyntöjä?
11. Voiko protokollan yli lähettää eri dataformaatteja?
12. Millä tavalla prosessoitu data palautetaan pyytäjälle?
13. Tukeeko protokolla vastausosoitteita?
14. Millä tavalla protokolla käsittelee virhetilanteet?

O-MI täytti kriteerit parhaiten, ja tarjosi jossain tapauksissa vaihtoehtoisia tapoja toteuttaa kriteeri. [10, s. 11–22] Tämä monipuolisuus ja sopeutuvuus eri käyttötilanteisiin tekee O-MI:sta hyvän kandidaatin esineiden internetin yleiseksi standardiksi, ja siksi se on valittu käytettäväksi tässä työssä älylaitteiden kanssa kommunikointiin. Luvussa 3 käsitellään O-MI:n ominaisuuksia tarkemmin.

O-MI on viestintäprotokolla, eikä se vaadi kuljettamansa tiedon olevan mis-
sään tietystä muodossa, mutta se on suunniteltu toimimaan yhdessä *Open*

Data Format -tiedostomuodon kanssa. Luku 3 käsittelee myös O-DF:ää tarkemmin.

O-MI on standardina varsin tuore, eikä sitä käytetä vielä paljon. Euroopan unionin rahoittama tutkimusprojekti bIoTope³ käyttää niitä perustana avoimien ekosysteemien luomiselle. BIoTope tulee englannin sanoista *building an IoT OPen innovation Ecosystem for connected smart objects* eli avoimien esineiden internetin innovaatioekosysteemin rakentaminen verkotetuille älylaitteille. Projekti pyrkii mahdollistamaan tiedon julkaisun ja käsittelyn monentyyppisistä lähteistä, mikä on tarve, johon O-DF ja O-MI pystyvät vastaamaan.

Kubler, Främling ja Derigent ovat tehneet tapaustutkimuksen O-DF/O-MI:n käyttämisestä kotiautomaatiossa ja terveydenhuollossa. Kotiautomaatiossa perhe hallitsee älypuhelimiltaan kodin jääkaappia ja ilmastointilaitetta. Terveydenhoitoesimerkissä hölkkääjästä kerätty terveystieto jaetaan sydänlääkärin ja vakuutusyhtiön kanssa. [29, s. 90–94] Standardeja on myös sovellettu rakentamisen hallintaan [8].

2.3.2 Standardi terveydenhuollon tietojärjestelmille

Terveydenhuollon tietojärjestelmissä Health Level 7 Internationalin (HL7) kehittämät standardit, kuten HL7 version 2 ja HL7 version 3, ovat yleisiä. HL7 version 3:ta käytetään esimerkiksi Yhdistyneen kuningaskunnan julkisessa terveydenhuollossa [24]. Uusin HL7:n kehittämä standardi on *Fast Healthcare Interoperability Resources* (FHIR). Sen tarkoitus on yhdistää edeltävien standardien parhaat puolet. Se käyttää muita internetin standardeja, kuten XML:ää ja HTTP:tä, ja tarjoaa RESTful-rajapinnan. [16]

FHIR:iä on sovellettu tutkimuksessa. Sitä on käytetty älylasien yhdistämiseen sairaalan tietojärjestelmään. Kun käyttäjä katsoo lasella potilasta tai esinettä, josta on järjestelmässä tietoa, lasit voivat näyttää tiedon. [34] Toinen sovelluskohde on ollut AidIT-sovellus, jonka tarkoitus on tehostaa potilastiedon käyttöä. Sovelluksen tekijät havaitsivat FHIR:n soveltuvan hyvin mobiilien terveydenhuollon sovellusten toteuttamiseen. Varsinkin sen RESTful-rajapinta sai kiitosta. [31]

Vaikka FHIR ei ole yleisessä käytössä, se luultavasti tulee yleistymään muiden HL7:n standardien tavoin. Tässä työssä toteuttavan järjestelmän halutaan ennakoivan tulevaisuuden järjestelmiä, joten vielä tuloillaan olevan standardin käyttö on perusteltua. Myöskin sen ominaisuudet, kuten XML-

³<http://biotope.cs.hut.fi/>

muotoinen data ja RESTful-rajapinta, mahdollistavat toimivan järjestelmän kehittämisen nopeammin. Luvussa 4 käydään läpi tarkemmin ne FHIR:n ominaisuudet, joita käytetään tässä työssä.

Luku 3

Open Data Format ja Open Messaging Interface -standardit

Tämä luku käsittelee Open Data Format (O-DF) ja Open Messaging Interface (O-MI) -standardeja, jotka on tarkoitettu toimimaan esineiden internetissä samoissa rooleissa, joissa HTML ja HTTP toimivat perinteisessä. The Open Group on julkaissut molemmat standardit. Luvussa 3.1 käydään läpi O-DF. Luku 3.2 käsittelee O-MI:tä.

3.1 Open Data Format

Open Data Format (O-DF) on dataformaatti, joka on tarkoitettu tiedon julkaisemiseen esineiden internetissä. Sen on tarkoitus kuvata tietoa esineistä standardoidulla tavalla, jota monet järjestelmät pystyvät ymmärtämään. [36, s. 1]

O-DF on määritelty *Extensible Markup Language* -skeemalla (XML-skeema). Formaatti perustuu oliohierarkiaan. Ylin elementti on Objects-elementti (oliot), jonka lapsina on muita olioita. Olioilla on yksi pakollinen kenttä, tunniste. [36, s. 1–2] Tunnisteiden tulisi olla globaalisti uniikkeja, mutta myös pelkkä ohjelman sisäinen yksilöllisyys riittää [36, s. 4]. Valinnaisesti oliolla voi olla kuvaus (description), sekä mielivaltainen määrä aliolioita ja InfoItem-kenttiä. Puun syvyyttä ei ole rajoitettu. [36, s. 1–2]

InfoItem-elementti kuvaa olion ominaisuuksia. Sen ainoa pakollinen attribuutti on nimi (name). Valinnaisia kenttiä ovat metadata, kuvaus ja arvo (value). Metadata kuvaa InfoItemin sisältämää tietoa, esimerkiksi arvokentissä käytettyä yksikköjä. Kuvauskenttä kertoo selkokielisesti, mitä tietoa In-

```
<Objects>
  <Object type="Sykemittari">
    <id>Sykemittari123ABC</id>
    <InfoItem name="sykemittaus">
      <description>Syke tietyllä ajanhetkellä
      </description>
      <MetaData>
        <InfoItem name="unit">
          <value type="xs:string">iskua
            minuutissa
          </value>
        </InfoItem>
      </MetaData>
      <value dateTime="2016-06-05T13:23:21">
        61
      </value>
      <value dateTime="2016-06-05T13:28:56">
        65
      </value>
    </InfoItem>
  </Object>
</Objects>
```

Listaus 3.1: Esimerkki O-DF-rakenteesta.

foItem sisältää. Arvokentissä on olion tietoarvoja, esimerkiksi anturien mittaustuloksia.

Listauksessa 3.1 on esimerkki O-DF-rakenteesta. Se kuvaa sykemittaria, jolla on kaksi mittaustulosta eri ajanhetkiltä. Uloin objects-elementti sisältää yhden olion, jota kuvaa Object-elementti. Sen attribuutti type (tyyppi) kertoo, minkälainen olio on kyseessä. Valinnaisesti type-attribuutti voi sisältää linkin luokkamäärittelyyn [36, s. 4].

Olio sisältää kaksi elementtiä: tunnisteen ja InfoItemin, joka kertoo sykkeen tietyllä ajanhetkellä. InfoItemin kuvauselementti kertoo luonnollisella kielellä, mistä InfoItemissa on kyse. Kaksi arvoelementtiä kertovat sykkeen, ja aika on talletettu niiden dateTime-attribuutteihin. InfoItemin MetaData-elementti kertoo, mikä arvoelementeissä käytetty yksikkö on.

Koska O-DF on vain dataformaatti, sillä kuvattua tietoa voidaan siirtää millä tahansa tavalla. O-DF:n määritelmä antaa esimerkin RESTful-rajapinnan

(Representational State Transfer) käytöstä [36, s. 8]. Toinen mahdollinen tapa on käyttää seuraavassa kappaleessa kuvattua Open Messaging Interfacea.

3.2 Open Messaging Interface

Open Messaging Interface (O-MI) on rajapinta elinkaaritiedon välittämiseen automaattisten järjestelmien välillä [37, s. 1]. Protokollapinossa se toimii useimpien alemman tason protokollien päällä. Esimerkiksi HTTP:tä, SMTP:tä tai tekstiviestejä voidaan käyttää O-MI-viestien kuljettamiseen. [37, s. 2] Tässä työssä sitä käytetään HTTP:n yli, koska HTTP on yleisin internetin protokolla. O-MI:n HTTP-toteutus perustuu POST ja GET-operaatioihin. POST-operaatiolla suoritetaan luku- kirjoitus ja peruutusviestit ja niiden vastaukset. GET-operaatiota käytetään tietolähteiden julkaisuun ja löytämiseen. [37, s. 5–6]

O-MI perustuu tarkkailijasunnittelumalliin [37, s. 5]. Siinä yksi olio on subjekti, jolle tarkkailijat ilmoittavat haluansa tietoja muutoksista. Kun subjektin tila muuttuu, se ilmoittaa kaikille tarkkailijoille siitä. [11, s. 293–295] O-MI-solmut voivat rekisteröidä itsensä tarkkailijoiksi muille O-MI-solmuille, jolloin ne saavat ilmoituksen muutoksista tai päivityksiä tilasta tietysin väliajoin. O-MI-solmut voivat toimia monessa roolissa yhtä aikaa, eli ne voivat yhtä aikaa tarkkailla muita solmuja ja ilmoittaa muille solmuille muutoksista. Niiden muodostama verkko on siis vertaisverkko, jossa jokainen solmu on sekä palvelin että asiakas. [37, s. 1]

O-MI:ssa on kolme operaatiota: luku, kirjoitus ja peruutus. Luvulla haetaan tietoa solmuista ja rekisteröidytään tarkkailijaksi. Kirjoituksella tehdään tietopäivityksiä muihin solmuihin. Peruutuksella lopetetaan toisen solmun tarkkailu ennen sen ennaltasovittua päättymistä. [37, s. 2] Kaikkia kolmea viestityyppiä tullaan käyttämään tässä työssä, joten ne kaikki selitetään tässä kappaleessa.

Viestin vastaanottajaksi voidaan merkitä jokin muu solmu kuin se, jolle viesti ensimmäisenä lähetetään. Tällöin muut solmut lähettävät viestin eteenpäin vastaanottajaksi merkitylle solmulle. Jos vastaanottajia ei ole eritelty, viestin vastaanottaja ei lähetä viestiä eteenpäin. [37, s. 9]

```
<?xml version="1.0" encoding="UTF-8"?>
<omi:omiEnvelope
  xmlns:xs="http://www.w3.org/2001/XMLSchema-
instance"
  xmlns:omi="omi.xsd"
  version="1.0"
  ttl="0">
  <omi:read msgformat="odf">
    <omi:msg>
      <Objects xmlns="odf.xsd"/>
    </omi:msg>
  </omi:read>
</omi:omiEnvelope>
```

Listaus 3.2: Esimerkki O-MI-kertalukuviestistä.

3.2.1 Kertaluku

Lukuviestillä on kaksi toimintoa: kertaluku ja tarkkailijaksi rekisteröityminen, jota O-MI-standardi kutsuu tilaukseksi. Kertaluvussa solmua pyydetään lähettämään senhetkinen tieto [37, s. 15]. Listauksessa 3.2 on esimerkki kertalukuviestistä, joka hakee kaikki O-DF Objects-elementin alaiset oliot. Uloin elementti *omi:omiEnvelope* on jokaisen O-MI-viestin uloin elementti, joka sisältää jonkin luku- kirjoitus-, peruutus- tai vastauselementeistä. Sen attribuuteissa määritellään käytetty O-MI-skeema, O-MI:n versionumero (kirjoitushetkellä on olemassa vain 1.0) sekä TTL (Time to Live), joka kertoo sekunneissa, kuinka kauan viesti on voimassa. Viesti, jonka TTL arvo on 0, on voimassa HTTP-yhteyden ollessa auki. Elementti *omi:read* määrittelee lukuviestin. Attribuutti *msgformat* kertoo, missä formaatissa viesti on; kuvan tapauksessa formaatti on O-DF. Alin elementti *omi:msg* sisältää itse viestin aiemmin määritellyssä formaatissa. [37, s. 8–9]

3.2.2 Tilausviesti

Tilausviesti kertoo vastaanottajalle, että lähettäjä haluaa tietoa tietyin väliajoin. Väliaika annetaan sekunneissa. Päivityksiä voi myös pyytää lähettäväksi aina, kun tieto muuttuu. Jos tilausviestissä on annettu vastausosoite, solmu lähettää pyydetyn tiedon sinne noudattaen pyydettyä väliaikaa mahdollisimman hyvin. Jos vastausosoitetta ei ole, solmu tallentaa pyydetyn tiedon, jotta se voidaan lähettää vastaukseksi tulevalle lukuviestille, jolla on sama tunnis-

```
<omi:omiEnvelope xmlns:xs="http://www.w3.org/2001/
  XMLSchema-instance" xmlns:omi="omi.xsd" version="
  1.0" ttl="0">
  <omi:read msgformat="odf" interval="-1"
    callback="http://palvelin.fi/vastaus">
    <omi:msg>
      <Objects xmlns="odf.xsd"/>
    </omi:msg>
  </omi:read>
</omi:omiEnvelope>
```

Listaus 3.3: Esimerkki O-MI-tilausviestistä.

te kuin tilausviestillä. [37, s. 9–10] Molemmissa tapauksissa tilauspyyntöön myös vastataan viestillä, joka kertoo tilauksen vastaanottamisesta ja antaa sen tilausnumeron tilauksen peruuttamista varten, mutta ei sisällä pyydettyä tietoa [37, s. 16].

Listauksessa 3.3 on esimerkki O-MI-tilausviestistä. Muodoltaan se on samankaltainen kuin kertalukukin. Tilausviestin siitä tekee `omi:read`-elementin `interval`-attribuutti, joka määrittää, kuinka usein päivityksiä halutaan. Arvo `-1` tarkoittaa, että päivitys halutaan aina arvon muuttuessa. `Callback`-attribuutti määrittää osoitteen, jonne vastaus halutaan. Selkokielisesti kuvan tilaus pyytää kaikki vastaanottajan O-DF-oliot, kun jokin niistä muuttuu, ja pyytää vastauksen lähetettäväksi osoitteeseen *http://palvelin.fi/vastaus*.

3.2.3 Peruutusviesti

Peruutusviesteillä perutaan jo tehtyjä tilauksia. Peruutuksen tekemiseen tarvitaan tilausviestin kuittauksen mukana tullut tilausnumero. [37, s. 12] Listauksessa 3.4 esitetään O-MI-peruutusviesti tilaukselle, jonka tilausnumero on 1. Tilausnumero asetetaan requestID-elementtiin, joka on cancel-elementissä.

```
<omi:omiEnvelope xmlns:xs="http://www.w3.org/2001/
  XMLSchema-instance"
                  xmlns:omi="omi.xsd"
                  version="1.0" ttl="0">
  <omi:cancel>
    <omi:requestID>1</omi:requestID>
  </omi:cancel>
</omi:omiEnvelope>
```

Listaus 3.4: Esimerkki O-MI-peruutuksesta.

3.2.4 Kirjoitusviesti

Kirjoitusviesteillä solmuille voidaan kirjoittaa uutta dataa. Vastaanottava solmu kuittaa kirjoituksen, kun se on onnistunut, tai lähettää virheviestin. [37, s. 10–11] Listaus 3.5 esittää esimerkin kirjoitusviestistä. Write-elementin attribuutti *msgformat* kertoo, missä muodossa viestin sisältö on. Msg-elementti sisältää itse viestin. O-MI-viestit voivat sisältää dataa missä tahansa muodossa.

```
<omi:omiEnvelope xmlns:xs="http://www.w3.org/2001/
  XMLSchema-instance" xmlns:omi="omi.xsd"
                  version="1.0" ttl="0">
  <omi:write msgformat="odf">
    <omi:msg>
      <!-- Data O-DF-muodossa -->
    </omi:msg>
  </omi:write>
</omi:omiEnvelope>
```

Listaus 3.5: Esimerkki O-MI-kirjoitusviestistä.

Luku 4

Fast Healthcare Interoperability Resources

Tämä luku käsittelee *Fast Healthcare Interoperability Resources* -standardia (FHIR-standardi, lausutaan kuten englannin sana *fire*). FHIR:n tarkoitus on parantaa terveydenhuollon tiedon saatavuutta, löydettävyyttä ja ymmärrettävyyttä, sekä rakenteistaa ja standardoida sen esitysmuoto, jotta sitä voidaan käsitellä paremmin tietokoneilla. [14] Tämä saavutetaan määrittelemällä datalle standardoitu esitysmuoto (XML, JSON) sekä RESTful-rajapinta.

FHIR on laaja standardi. Tämän työn rajauksessa siitä tärkeät osat ovat datan esitysmuoto (alaluku 4.2), sekä palvelimen rajapinta (alaluku 4.3). Lisäksi alaluvussa 4.1 käsitellään standardin nykytilaa.

4.1 Standardin tila

FHIR käyttää kolmiportaista asteikkoa standardin valmiuden kuvaamiseen:

1. Luonnos: standardi ei ole tarpeeksi valmis toteuttavaksi.
2. Standardiluonnos kokeilukäyttöön (draft standard for trial use, DSTU): standardi on hyväksytty ja valmis toteutettavaksi, mutta se ei ole käytössä suuressa osassa tarkoitettuja käyttöympäristöjä.
3. Normatiivinen: Standardi on tarkastettu ja toteutettu monissa käyttöympäristöissä. Mahdolliset muutokset ovat pieniä, ja versioiden välinen yhteensopivuus on tärkeää.[15]

Standardin osat voivat olla alemmalla valmiustasolla kuin standardi kokonaisuudessaan, mutta toisinpäin se ei ole mahdollista. Esimerkiksi normatiivinen standardi voi sisältää luonnostilaisia osia, mutta DSTU-tason luonnos ei voi sisältää normatiivisia osia. [15]

FHIR on valmiustasoltaan DSTU 2 kirjoitushetkellä, eli toinen standariluonnos kokeilukäyttöön. Hackensackin yliopisto Yhdysvalloissa kehittää tietotekniikkayritys Inforin kanssa FHIR:a käyttävää järjestelmää DSTU 2:n pohjalta [25].

Valmiusasteikon lisäksi FHIR:llä on versionumero, joka perustuu semanttiseen versiointiin: ensimmäinen numero on julkaisu (esim. DSTU 1 oli versio 0, DSTU 2 on versio 1 jne.); toinen numero kasvaa, kun muutos rikkoo yhteensopivuuden; kolmas numero on pieni muutos, joka ei riko yhteensopivuutta; mahdollinen neljäs numero on revisio, joka muuttuu aina, kun spesifikaatiota muutetaan. Työn tekohetkellä 1.0.2 on uusin versio. [23]

Standardista tehdään uusi julkaisu 18–24 kuukauden välein. Seuraava julkaisu on vuonna 2017, ja sen on tarkoitus nostaa standardi normatiiviseksi ydinosaan. [15]

4.2 Resurssit

Resurssi FHIR:ssa on entiteetti, jota käytetään tiedon välittämiseen ja tallentamiseen. Se on instanssi jostain FHIR:ssa määritellyistä resurssityypeistä (potilas, mittaustulos ym.). Se sisältää rakenteellista dataa tyyppimäärittelynsä mukaisesti. Jokaisella resurssilla on myös yksilöivä tunnistus sekä versionumero, joka muuttuu aina resurssin muuttuessa. [13]

FHIR-standardi esittää XML:n ja JSON-muodot datalle. Myös muut esitysmuodot ovat sallittuja, mutta niille ei ole annettu esimerkkitehtäviä. [18]

Tässä työssä käytetään potilas-, havainto- ja laiteresursseja.

4.2.1 Potilasresurssi

```
<Patient xmlns="http://hl7.org/fhir">
  <id value="example"/>
  <text>
    <status value="generated"/>
    <!-- Selkokielineen kuvaus ihmislukijalle
          resurssista -->
  </text>
  <active value="true"/>
  <name>
    <use value="official"/>
    <family value="Chalmers"/>
    <given value="Peter"/>
    <given value="James"/>
  </name>
  <name>
    <use value="usual"/>
    <given value="Jim"/>
  </name>
</Patient>
```

Listaus 4.1: Esimerkki XML-muotoisesta potilasresurssista.

Potilaan henkilötietoja kuvataan Patient-resurssilla. Se sisältää nimen, syntymäajan ja osoitteen kaltaisia perustietoja. Potilaasta tehty mittaukset, diagnoosit ja annetut hoidot kuvataan muilla resursseilla. [20]

Listauksessa 4.1 on esitetty osa potilasresurssista. Juurielementti Patient asetetaan FHIR-nimiavaruuteen xmlns-attribuutilla. Id-elementti sisältää resurssin loogisen tunnisteen, joka on uniikki kyseiselle resurssityypille sillä palvelimella, jolla on data on. Globaalit tunnisteen menevät identifier-elementin alle, jota ei ole listauksessa esitetty. Name-elementtejä on kaksi, joista ensimmäinen on virallinen nimi, jota käytetään asiakirjoissa, ja toinen on puhutelnimi, jota käytetään potilaan kanssa keskustellessa.

4.2.2 Havaintoresurssi

```
<Observation xmlns="http://hl7.org/fhir">
  <id value="example"/>
  <category>
    <coding>
      <system value="http://hl7.org/fhir/observation-category"/>
      <code value="vital-signs"/>
      <display value="Vital Signs"/>
    </coding>
  </category>
  <code>
    <coding>
      <system value="http://loinc.org"/>
      <code value="3141-9"/>
      <display value="Weight Measured"/>
    </coding>
  </code>
  <subject>
    <reference value="Patient/example"/>
  </subject>
  <valueQuantity>
    <value value="80"/>
    <unit value="kg"/>
    <system value="http://unitsofmeasure.org"/>
    <code value="[kg]"/>
  </valueQuantity>
</Observation>
```

Lista 4.2: Esimerkki havaintoresurssista.

Mittaustuloksia esitetään Observation-resurssilla [19]. Kuvassa 4.2 on esimerkki punnitustuloksesta. Subject-elementti yhdistää punnitustuloksen potilaaseen. Value-attribuutin ensimmäinen osa kertoo resurssin, ja toinen osa resurssin id-elementin arvon. Esimerkin punnitustulos liittyy siis edellisen luvun potilaaseen.

Category (kategoria) kertoo, minkätyyppinen mitta on kyseessä. Alaelementti code on koneluettava arvo, kun taas display on tarkoitettu ihmisten luettavaksi. Tämä mittaustulos koskee potilaan peruselintoimintoja.

Kategorian jälkeinen code-elementti määrittää tarkalleen, mitä on mitattu. System ja code viittavat *Logical Observation Identifiers Names and Codes* -koodistoon, joka on terveydenhuollon käyttämä yleinen koodisto laboratorio- ja kliinisille tuloksille. Tässä tapauksessa mitataan potilaan paino. Viimeinen elementti ValueQuantity kertoo tuloksen: tässä tapauksessa 80 kilogrammaa.

```
<Device xmlns="http://hl7.org/fhir">
  <id value="example-pacemaker"/>
  <identifier>
    <system value="http://acme.com/devices/
      pacemakers/octane/serial"/>
    <value value="1234-5678-90AB-CDEF"/>
  </identifier>
  <type>
    <coding>
      <system value="http://acme.com/devices"/>
      <code value="octane2014"/>
      <display value="Performance_pace_maker_for
        high_octane_patients"/>
    </coding>
  </type>
  <manufacturer value="Acme_Devices,_Inc"/>
  <model value="PM/Octane_2014"/>
  <lotNumber value="1234-5678"/>
  <patient>
    <reference value="Patient/example"/>
  </patient>
</Device>
```

Listaus 4.3: Sydämentahdistimen FHIR-laiteresurssi.

4.2.3 Laiteresurssi

FHIR-laiteresurssi kuvaa terveydenhuollossa käytettyä laitetta. Resurssi sisältää sekä lääketieteelliset laitteet, kuten verenpainemittarit ja älyimplantit että muut laitteet (tietokoneet, puhelimet). [17] Laiteresurssia tullaan käyttämään toteuttavassa järjestelmässä edustamaan mittaavia laitteita.

Listauksessa 4.3 on sydämentahdistimen laiteresurssi. Rakenteeltaan se on samankaltainen kuin edelliset resurssit. Laitteelle voidaan antaa valmistaja,

malli ja sarjanumero. Se voidaan yhdistää potilaaseen Patient-elementillä, jos laiteella on vain yksi käyttäjä.

4.3 RESTful-rajapinta

FHIR-palvelinten kanssa kommunikoidaan RESTful-rajapinnalla. Jos FHIR-palvelin sijaitsee osoitteessa *https://palvelin.fi*, niin luvussa 4.2.1 kuvattu potilasresurssi olisi haettavissa tekemällä HTTP GET-pyyntö osoitteeseen *www.palvelin.fi/Patient/example*. Resurssin id-elementti määrittää osoitteen viimeisen osan. Kyseistä potilasresurssia voidaan päivittää tekemällä HTTP PUT-pyyntö samaan osoitteeseen. Uusi potilasresurssi luotaisiin tekemällä HTTP POST-pyyntö osoitteeseen *https://www.palvelin.fi/Patient*. [21]

Rajapinta mahdollistaa myös hakujen tekemisen millä tahansa resurssien sisältämällä kentällä. Haku suoritetaan tekemällä GET-pyyntö resurssille ja antamalla pyynnön parametreiksi halutut hakuehdot. Esimerkiksi kaikki potilaat, joiden sukunimi on Potilas, voidaan löytää antamalla parametri *name="Potilas"*. [22] Hakuominaisuutta tullaan käyttämään toteuttavassa järjestelmässä löytämään mittaustuloksiin liittyvän potilaan ja laitteen vastinparit FHIR-palvelimelta.

Luku 5

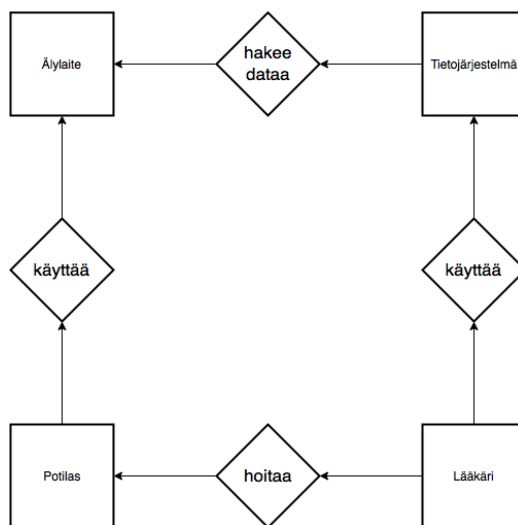
Toteutus

Tässä työssä tutkitaan O-MI / O-DF -tekniikoiden yhteensovittamista FHIR-standardia toteuttavien tietojärjestelmien kanssa. Tutkimus toteutetaan tekemällä prototyyppi kotihoidon järjestelmästä, jossa joukko terveydenhuollon älylaitteita lähettää O-DF-muotoista dataa O-MI:n yli FHIR-muunnospalvelimelle, joka muuntaa datan FHIR-muotoon ja lähettää sen eteenpäin varsinaiselle FHIR-palvelimelle. Tutkimuksen tulos on dokumentaatio siitä, miten kuvatulainen järjestelmä toteutetaan ja miten FHIR-resursseja kuvataan O-DF-muodossa.

Luvussa 5.1 selitetään järjestelmän käyttöympäristö. Sitä seuraavassa luvussa 5.2 esitellään järjestelmän mahdolliset toteutusvaihtoehdot, niiden edut ja haitat ja niistä valitaan yksi toteutettavaksi. Luku 5.3 kuvaa toteutettavan järjestelmän ja erittelee sen osat. Luvussa 5.4 kerrotaan, mitä on rajattu toteutuksen ulkopuolelle ja mistä syistä. Viimeisessä luvussa 5.5 kerrotaan, miten toteutettua järjestelmää voi kokeilla.

5.1 Skenaario

Työssä toteutettu järjestelmä simuloi kotihoitoa. Järjestelmää käyttävä potilas sairastaa sydän- ja verisuonitauteja. Lääkäri on ohjeistanut häntä pudottamaan painoaan ja alentamaan verenpainettaan. Jotta lääkäri voi seurata potilaan edistymistä, potilaan älylaitteet kytketään lääkärin tietojärjestelmään. Luvussa 5.3.1 on eritelty simuloidut laitteet. Potilas on perheellinen, ja hänen perheenjäsenensä voivat myös käyttää laitteita. Kuva 5.1 havainnollistaa skenaarion.



Kuva 5.1: Skenaarion ER-kaavio.

Kotihoidon skenaario valittiin sen yksinkertaisen toteutuksen vuoksi: yhdellä potilaalla ei ole käytössään montaa laitetta, ja laitteet ja niiden mittaustulokset ovat yksinkertaisia ymmärtää terveydenhuoltoon perehtymättömälle. Toteutuksessa käytetyt esimerkkilaitteet ja -toimintatavat eivät välttämättä vastaa todellisia terveydenhuollon laitteita eivätkä toimintatapoja. Koska tämä työ keskittyy järjestelmän teknisiin osiin, mahdollinen epäyhteensopivuus todellisiin toimintatapoihin ei ole ongelma.

5.2 Toteutusvaihtoehdot

Skenaariossa käytettävän järjestelmän voi toteuttaa hyödyntäen tässä työssä esiteltyjä tekniikoita eri tavoilla. Tässä luvussa käydään läpi kolme vaihtoehtoa: puhdas FHIR-järjestelmä, puhdas O-DF/O-MI-järjestelmä sekä molempia tekniikoita hyödyntävä sekajärjestelmä.

5.2.1 Puhdas FHIR-järjestelmä

Puhtaassa FHIR-järjestelmässä älylaitteet toteuttavat FHIR-standardin, eli ne pystyvät lähettämään FHIR-resursseja XML- tai JSON-muodossa. Tällöin ne voivat suoraan kommunikoida FHIR-tietojärjestelmän kanssa. Kommunikointiin käytettäisiin esimerkiksi normaalia HTTP:tä. Kuten kuvassa 5.2 on esitetty, tällainen järjestelmä on korkealla tasolla hyvin yksinkertainen.

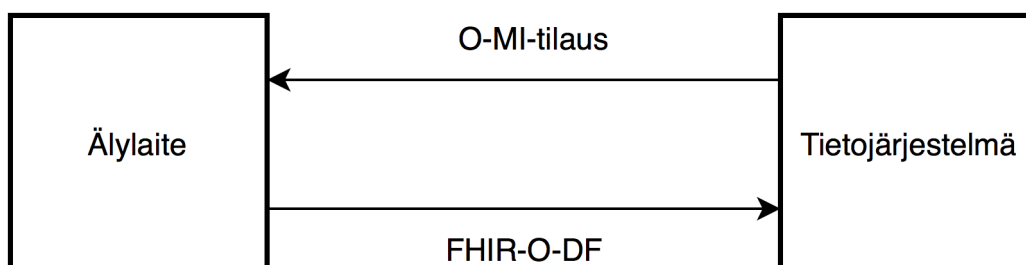


Kuva 5.2: Pelkkää FHIR:ä käyttävä järjestelmä.

Suurin etu puhtaassa FHIR-järjestelmässä on, että datan muotoa ei tarvitse muuttaa missään vaiheessa. Haitta on, että tarjottu data on sidottu terveydenhuollon kontekstiin. Jos dataa halutaan käyttää johonkin muuhun tarkoitukseen, se pitää muuntaa. Tämä ei ole ongelma, jos laite on tarkoitettu vain terveydenhuoltoon, kuten esimerkiksi lääkärin potilaalle hoidon ajaksi antama laite. Sen sijaan yleiskäyttöisemmän laitteen, kuten älykellon, dataa voi käyttää moni järjestelmä. Pahimmassa tapauksessa laite voi joutua pyörittämään montaa ohjelmaa, jotta se voi tarjota samaa dataa eri järjestelmiin. Parasta olisi, että laitteen tarjoama data ei ottaisi kantaa käyttötarkoitukseensa, eli sitä ei kannata esittää laitteessa FHIR:nä.

5.2.2 Puhdas O-DF/O-MI-järjestelmä

Puhtaassa O-DF/O-MI-järjestelmässä myös tietojärjestelmä on O-MI-noodi. Se voi suoraan tilata päivitykset älylaitteilta O-MI-standardin mukaisesti. FHIR-standardissa resursseja on esitetty XML- ja JSON-muodoissa, mutta niitä voi esittää myös muissa muodoissa, kuten O-DF:nä. Esittämällä kaikki resurssit FHIR-O-DF:nä datan muoto voidaan pitää samana koko järjestelmän sisällä. Lisäksi O-MI:n käyttö tarjoaa standardoidun tavan pyytää tietoa. Kuvassa 5.3 on esitetty tämä toteutusmalli.

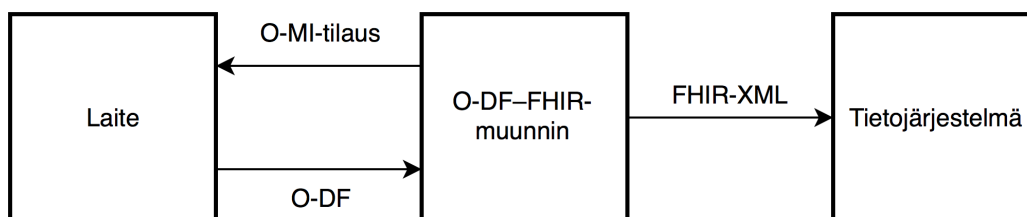


Kuva 5.3: Puhdas O-DF/O-MI-järjestelmä.

Tämän toteutusvaihtoehdon ongelma on tietojärjestelmän yhteensopivuus muiden FHIR-järjestelmien kanssa. Ne eivät todennäköisesti tue O-DF-muotoista FHIR:ä, jolloin resurssit täytyy konvertoida yleisempiin XML- tai JSON-muotoihin. Jos konversio täytyy joka tapauksessa tehdä, FHIR:n O-DF-muodon toteuttaminen on turhaa työtä. FHIR-O-DF:llä on myös sama ongelma kuin edellisen luvun toteutuksella: data sitoutuu terveydenhuollon kontekstiin, mikä vähentää sen käytettävyyttä muihin tarkoituksiin.

5.2.3 Sekajärjestelmä

Sekajärjestelmä hyödyntää sekä O-DF/O-MI:a että FHIR:iä. Älylaitteet ovat O-MI-noodeja, jotka lähettävät tietonsa eteenpäin O-DF–FHIR-muuntimelle, joka myös on O-MI-noodi. Se muuntaa datan FHIR:n XML-muotoon ja lähettää sen palvelimelle. Kuva 5.4 havainnollistaa järjestelmän toiminnan.



Kuva 5.4: Sekajärjestelmä.

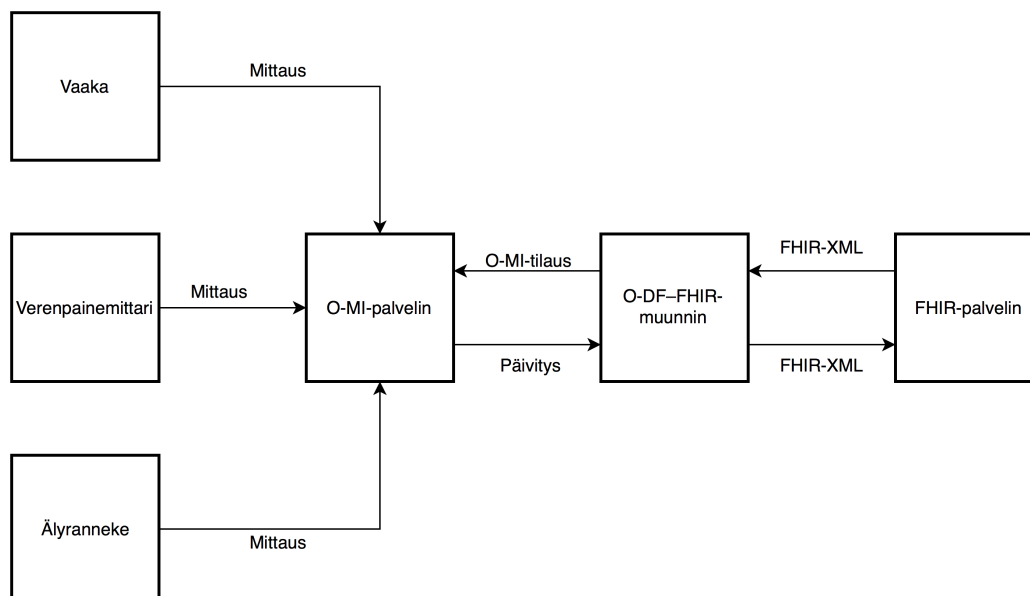
Tämä ratkaisu välttää kahden edellisen vaihtoehdon ongelmat: älylaitteet voivat helposti toimia yhdessä ei-FHIR-järjestelmien kanssa, ja FHIR-järjestelmä toteuttaa standardin yleiset osat, jolloin se sopii yhteen muiden FHIR-järjestelmien kanssa. Muunnin myös mahdollistaa taustalla olevan FHIR:ä toteuttavan järjestelmän vaihtamisen toiseen. Näistä syistä sekajärjestelmä valittiin toteutettavaksi.

5.3 Järjestelmän osat

Järjestelmä koostuu neljästä osasta:

1. Älylaitteet
2. O-MI-palvelin
3. O-DF-FHIR-muunnin
4. FHIR-palvelin

Kuvassa 5.5 on havainnollistettu järjestelmän arkkitehtuuri. Älylaitteita on kolme: vaaka, verenpainemittari ja älyranneke. Luku 5.3.1 esittelee ne tarkemmin. Ne tekevät mittauksia ja lähettävät ne O-DF-muodossa O-MI-palvelimelle. O-DF-FHIR-muunnin tekee tilauksen älylaitteiden datasta O-MI-palvelimelle. Palvelin lähettää laitteiden dataa eteenpäin aina, kun uusi päivitys saapuu. Muunnin käsittelee datan FHIR:n XML-muotoon ja lähettää sen HTTP POST-pyyntönä FHIR-palvelimelle, joka tallentaa tiedon kantaan. Jotta muunnin voi linkittää FHIR-resursseja toisiinsa, se hakee niitä ajoittain palvelimelta.



Kuva 5.5: Järjestelmän osat ja niiden yhteydet.

Erillinen O-MI-palvelin ei ole pakollinen osa järjestelmää: laitteet voisivat olla itse O-MI-solmuja ja kommunikoida muuntimen kanssa. Tässä työssä

käytetään erillistä palvelinta kehityksen helpottamiseksi: O-MI:n toteuttava referenssipalvelin on jo olemassa, joten käyttämällä sitä O-MI:a ei tarvitse toteuttaa laitteille erikseen.

5.3.1 Älylaitteet

Älyesineet esittävät moderneja laitteita, jotka pystyvät kommunikoimaan toistensa ja muiden järjestelmien kanssa. Tässä työssä käytetään kolmea laitetta: vaakaa, älyranneketta ja verenpainemittaria. Valitettavasti työtä varten ei saatu käyttöön oikeita laitteita, joten laitteita simuloidaan Java-ohjelmalla. Se arpoo 10 sekunnin välein laitteilta yhden mittaustuloksen lähetettäväksi O-MI-palvelimelle. Taulukossa 5.1 on esitetty simuloidut laitteet ja niiden mittaamat suuret.

Suure	Mittaava laite	Yksikkö
Paino	Vaaka	kilogramma
Rasvaprosentti	Vaaka	prosentti
Verenpaine	Verenpainemittari	mmHg
Askeleet	Älyranneke	Askeleet, kpl
Energiankulutus	Älyranneke	kilojoule
Syke	Älyranneke	iskua / min
Yöunet	Älyranneke	minuutti

Taulukko 5.1: Laitteiden mittaamat suuret.

5.3.1.1 Vaaka



Kuva 5.6: Withings Smart Body Analyzer & Wireless Scale -henkilövaaka.

Vaaka esittää markkinoilla olevia älyvaakoja. Esimerkiksi Withings, FitBit ja Polar valmistavat tällaisia laitteita. Kuvassa 5.6¹ on Withingsin valmistama malli. Vaaka punnitsee henkilön massan kilogrammoissa ja mittaa arvion henkilön rasvaprosentista.

¹Kuvan tekijänoikeus Dai44, 2014. Lisensoitu Creative Commons Attribution 2.0 Generic. Kuvan lähde <https://www.flickr.com/photos/70508773@N00/14063822404>

5.3.1.2 Älyranneke



Kuva 5.7: Withings Pulse O2 Fitness Tracker.

Älyranneke simuloi 2010-luvulla yleistyneitä älykelloja ja muita ranteessa pidettäviä laitteita, kuten esimerkiksi Fitbit, Apple Watch ja Samsung Galaxy Gear. Kuvassa 5.7² on Withingsin Pulse O2 Fitness Tracker.

Seuraavia mittaustuloksia simuloidaan:

- Askelmittari
- Energiankulutus
- Syke
- Yöunet

Mittaustulokset ovat erillisiä, eivätkä ne generoidu yhtä aikaa. Niistä voidaan tilata vain osa.

²Kuvan tekijänoikeus Gary J Wood, 2014. Lisensoitu Creative Commons Attribution-ShareAlike 2.0 Generic. Kuvan lähde <https://www.flickr.com/photos/garyjwood/15044532631/>

5.3.1.3 Verenpainemittari

Modernit verenpainemittarit voivat lähettää tuloksensa käyttäjän laitteille. Esimerkiksi kuvassa 5.8³ esitetty Withingsin verenpainemittari voi lähettää tuloksensa Android- ja iOS-puhelimille. Tämän työn verenpainemittari esittää tällaista laitetta. Se mittaa ylä- ja alapaineen, mutta ne lähetetään eteenpäin erillisinä tietoina. Tämä johtuu terveydenhuollon käyttämistä mittaus-
ten tunnistuskoodeista, jotka erottavat paineet toisistaan. Luvussa 6.2 kerrotaan näistä koodeista tarkemmin.



Kuva 5.8: Withings Blood Pressure Monitor -verenpainemittari.

5.3.2 O-MI-palvelin

O-MI-palvelin kerää datan kodin älylaitteilta ja mahdollistaa siihen käsiksi-pääsyn kodin ulkopuolelta. Tässä työssä käytettiin palvelimena Aalto-ylio-

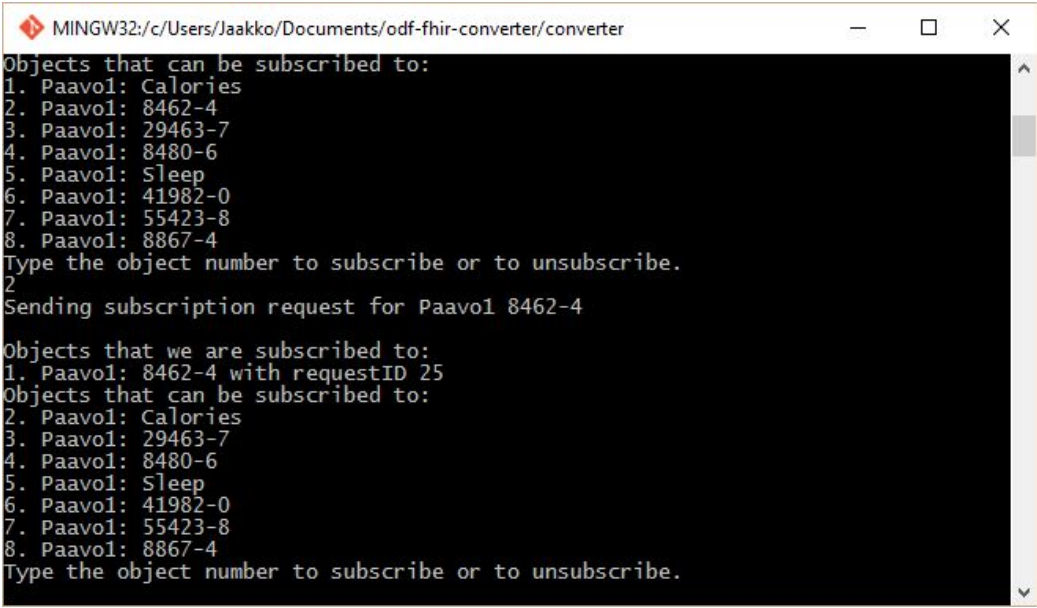
³Kuvan tekijänoikeus Code of the new, 2014. Lisensoitu Creative Commons Attribution 2.0 Generic. Kuvan lähde <https://www.flickr.com/photos/75654647@N05/15010985868>

piston referenssipalvelinta⁴, joka toteuttaa koko O-MI-standardin. Laitteita simuloiva Java-ohjelma lähettää sille O-MI-kirjoituspyyntöinä päivitykset.

Kuten aiemmin todettiin, järjestelmän kannalta O-MI-palvelin ei ole välttämätön. Laitteet voisivat olla itse O-MI-solmuja, joista data pyydetään. Erilisen palvelimen käyttämisellä on kuitenkin etunsa: se mahdollistaa kaiken yhteen henkilöön liittyvän datan hakemisen yhdestä paikasta. Tulokappaleessa 6.1 kerrotaan tarkemmin, mitä vaikutuksia O-MI-solmujen määrällä on datan esitysmuotoon.

5.3.3 O-DF–FHIR-muunnin

O-DF–FHIR-muuntimen tehtävä on kääntää O-MI:n yli tulevat O-DF-muotoiset viestit FHIR:n XML-muotoon ja lähettää ne HTTP:n yli FHIR-palvelimelle. Tässä työssä toteutettu muunnin ei toteuta O-MI:ä täysin: se pystyy lukemaan muita solmuja sekä tekemään tilauksia ja perumaan niitä, mutta siltä itseltään ei voi lukea mitään.



```
MINGW32: c:/Users/Jaakko/Documents/odf-fhir-converter/converter
Objects that can be subscribed to:
1. Paavo1: Calories
2. Paavo1: 8462-4
3. Paavo1: 29463-7
4. Paavo1: 8480-6
5. Paavo1: Sleep
6. Paavo1: 41982-0
7. Paavo1: 55423-8
8. Paavo1: 8867-4
Type the object number to subscribe or to unsubscribe.
2
Sending subscription request for Paavo1 8462-4
Objects that we are subscribed to:
1. Paavo1: 8462-4 with requestID 25
Objects that can be subscribed to:
2. Paavo1: Calories
3. Paavo1: 29463-7
4. Paavo1: 8480-6
5. Paavo1: Sleep
6. Paavo1: 41982-0
7. Paavo1: 55423-8
8. Paavo1: 8867-4
Type the object number to subscribe or to unsubscribe.
```

Kuva 5.9: Toteutetun muuntimen komentorivikäyttöliittymä.

Skenaariossa muunnin toimii myös lääkärin käyttöliittymänä. Lääkäri voi muuntimen käyttöliittymästä nähdä, mitä dataa potilas julkaisee, ja tilata hoidon kannalta tärkeät tiedot. Kuvassa 5.9 on muuntimelle toteutettu

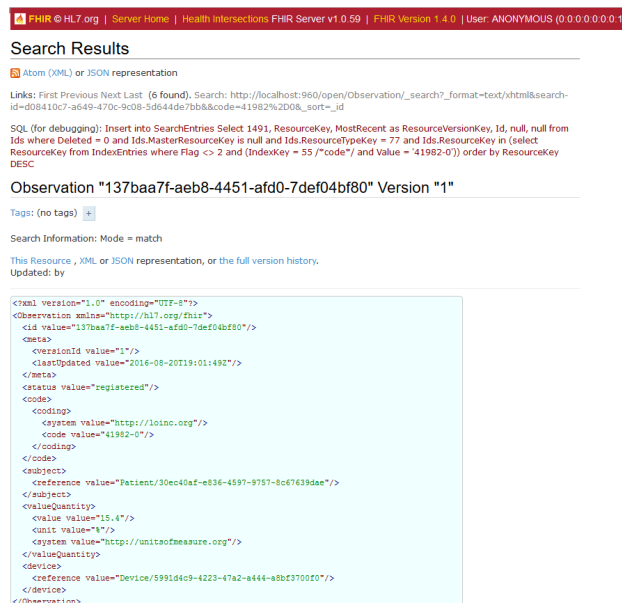
⁴<https://github.com/AaltoAsia/O-MI>

yksinkertainen käyttöliittymä. Käyttäjä voi numeroita syöttämällä tilata ja peruuttaa mittauksia. Paavo1 on potilaan tunniste, ja numerosarjat ovat lääketieteessä käytettyjä havaintojen tunnistekoodoja.

Kun toteutettu muunnin aloittaa toimintansa, se hakee argumenttina annetun potilaan O-DF-olion palvelimelta. Jos potilasta ei ole vielä olemassa FHIR-palvelimella, se lisää sen. Tämän jälkeen se esittää listan kaikista potilaaseen liittyvistä mittauksista, joita voi tilata. Kun tilaus tehdään, O-MI-palvelin alkaa lähettää päivityksiä aina, kun laite niitä tekee. Lähetetyssä tiedossa on vain mittauksen tulos, joten muuntimen pitää hakea FHIR-havainnon tekemiseksi lisätietoa sekä O-MI- että FHIR-palvelimelta. Kun havainto on muunnettu, se lähetetään FHIR-palvelimelle.

5.3.4 FHIR-palvelin

FHIR-palvelimena käytetään Graham Grieben referenssitoteutusta⁵. Se on geneerinen FHIR-palvelimen toteutus SQL-tietokantaa vastaan. Tietokantana on Microsoftin SQL Server 2016 Express.



Kuva 5.10: Vaa'an tuottama rasvaprosenttimittaus muunnettuna FHIR-palvelimella.

⁵<https://github.com/grahamegrieve/fhirserver>

Tässä työssä FHIR-palvelimen tehtävä oli validoida tuotetut FHIR-resurssit standardin mukaisiksi. Kuvassa 5.10 on vaa'an tuottama rasvaprosenttimittaus FHIR-palvelimen selainkäyttöliittymästä tarkasteltuna.

5.4 Rajaukset

Järjestelmä on toteutettu soveltuvuusselvityksenä, joten tehokkuutta ei ole otettu huomioon. Oikeassa järjestelmässä muuntimen tehokas toiminta on tärkeää, varsinkin jos kotihoidettavien potilaiden määrä kasvaa.

Järjestelmän tietoturva on rajattu työn ulkopuolelle. Terveystilaan liittyvä tieto on Suomen lainsäädännössä luokiteltu arkaluonteiseksi tiedoksi⁶, mikä tarkoittaa, sen käsittelyyn tarvitaan lupa ja sitä saa käsitellä vain valtuutetut henkilöt. Oikeaa dataa käsittelevän järjestelmän pitää täyttää lain vaatimukset. Koska tässä työssä ei käsitellä oikeaa dataa, eikä tietoturvan toteuttaminen ole oleellista tutkimuskysymyksiin vastaamiseksi, järjestelmälle ei ole toteutettu minkäänlaista tietoturvaa. Tietoturvallista O-MI:n käyttöä on tutkittu omassa diplomityössään [10].

5.5 Lähdekoodi

Kaikki tätä työtä varten kirjoitettu koodi on saatavilla Git-repositoriossa osoitteessa <https://bitbucket.org/Aakkosti/odf-fhir-converter>. Readme-tiedostoissa ohjeistetaan, miten järjestelmä asennetaan. Repositorion koodin lisäksi järjestelmän käyttämiseksi tarvitaan FHIR- ja O-MI-palvelimet. Koodi on testattu Aallon ja Grieben referenssipalvelimia vastaan, mutta sen pitäisi toimia kaikilla standardien mukaisilla palvelimilla.

⁶Henkilötietolaki, 3. luku, 11§

Luku 6

Tulokset

Tämä luku käsittelee järjestelmän toteutuksen aikana havaittuja ongelmia ja niiden ratkaisuja. Alaluvussa 6.1 kerrotaan, minkälainen O-DF-tietomalli valittiin ja miksi. Luvussa 6.2 käsitellään kenttien tunnistamisen ongelmaa. Sitä seuraava luku 6.3 selittää, miten älylaitteille luodaan niitä kuvaavat FHIR-resurssit. Kappale 6.4 käsittelee mittausten yksiköiden hakemista ja tunnistamista. Viimeinen alaluku 6.5 kertoo, miten O-DF-olioiden keskinäiset viittaukset toisinnetaan FHIR-resursseissa.

6.1 O-DF-datan esitysmuodon valinta

Järjestelmän älylaitteiden tulee julkaista mittaustuloksensa loogisessa rakenteessa, jotta sen lukeminen on mahdollisimman helppoa. Vaihtoehtoja on kaksi: laitelähtöinen ja käyttäjälähtöinen. Laitelähtöisessä rakenteessa juurioliot ovat laitteita, joilla on mittaustuloksia. Tulokset puolestaan linkittyvät käyttäjäolioihin. Listaus 6.1 esittää datan tällä rakenteella. Juuriolio tunnisteella SmartBand1 on älyranneke. Sillä on aliolio jokaiselle mittaustyypille, jonka se pystyy tekemään. Mittauksella on kaksi InfoItemia, joista toinen on linkki käyttäjäolioon, ja toinen on mittaustulos. Jos samasta käyttäjästä tehdään uusi mittaustulos, vain mittaustulos päivittyy. Jos toinen käyttäjä käyttää laitetta, molemmat kentät päivittyvät.

```
<Object>
  <id>SmartBand1</id>
  <!-- Tiedot laitteesta InfoItemeinä -->
  <Object>
    <id>8867-4</id> <!-- LOINC-koodi, syke -->
    <InfoItem name="user">
      <value type="xs:anyURI">
        http://10.10.10.10/Objects/Patient1
      </value>
    </InfoItem>
    <InfoItem name="measurement">
      <!-- Mittausdata -->
    </InfoItem>
  </Object>
</Object>
```

Listaus 6.1: Esimerkki laitelähtöisestä sykemittauksesta.

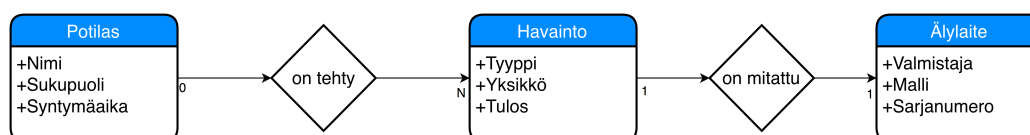
Käyttäjälähtöisessä rakenteessa käyttäjä on juuriolio. Oliolla on eri mittauksia alaolioina, jotka linkittyvät mitanneeseen laitteeseen. Listauksessa 6.2 on listauksessa 6.1 esitetyn laitelähtöisen rakenteen potilaslähtöinen versio. Ero- na aiempaan juuriolio on potilas, ja mittautulokset linkittyvät laitteeseen.

```
<Object>
  <id>Patient1</id>
  <!-- Tiedot potilaasta InfoItemeinä -->
  <Object>
    <id>8867-4</id> <!-- LOINC-koodi, syke -->
    <InfoItem name="device">
      <value type="xs:anyURI">
        http://10.10.10.10/Objects/
        SmartBand1</value>
      </InfoItem>
    <InfoItem name="measurement">
      <!-- Mittausdata -->
    </InfoItem>
  </Object>
</Object>
```

Listaus 6.2: Esimerkki potilaslähtöisestä sykemittauksesta.

Vaihtoehtoista käyttäjälähtöinen on parempi tämän työn tarkoitukseen, sillä olemme kiinnostuneita potilaaseen liittyvästä datasta, emme niinkään laitteesta, joka niitä on mitannut. Valitun rakenteen etu on, että käyttäjä voi vaihtaa mittaavaa laitetta ilman, että datan sijainti muuttuu. Listauksen 6.2 syketiedon saa aina URI:sta Patient1/8867-4/measurement. Jos rakenne olisi laitelähtöinen, URI muuttuisi laitteen vaihtuessa. Toinen etu on, että jos laite tekee mittauksen jostain muusta henkilöstä, sen tulos tallennetaan toisen olion alle. Simuloiduista laitteista vaaka ja verenpainemittari ovat helposti muiden kuin potilaan käytettävissä. Käyttäjälähtöisellä datalla O-MI-solmun on helpompi vastata pyyntöihin tietoturvallisesti: solmun asiakkaille voidaan antaa oikeuksia olion polun mukaan. Sen sijaan laitelähtöisessä pitäisi tarkistaa, mihin potilaaseen mittaus milloinkin liittyy, mikä on vaikeampaa.

Tässä työssä käytettiin erillistä O-MI-palvelinta, joka kokosi kaikki potilaaseen liittyvän tiedon yhteen O-MI-solmuun. Jos kaikki laitteet olisivat omia O-MI-solmuja, potilaaseen liittyvä data hajautuisi moneen paikkaan. Tässä tapauksessa käyttäjätieto pitäisi joko toisintaa jokaiselle solmulle, mikä on tiedon eheyden kannalta huono idea, tai siihen pitäisi linkittää. Ongelmallista olisi myös tiedon löytäminen: tiedon hakijan pitäisi tietää, millä solmuilla on tietoja käyttäjästä, tai pitäisi olla mahdollisuus tehdä hakuja. Onkin siten viisasta, että esimerkiksi kodilla on emäpalvelin, joka kokoaa tiedon kaikista kodin älylaitteista yhteen, vaikka laitteet olisivatkin itsenäisiä O-MI-solmuja. Tällöin kaikki henkilöstä, rakennuksesta ym. olevat tiedot ovat helposti löydettävissä.



Kuva 6.1: O-DF-datan ER-malli

Kuva 6.1 havainnollistaa toteutettua O-DF:n tietomallia ER-kaaviolla. Suorakulmiot on esitetty O-DF:ssä olioina, plus-merkit ja relaatiot InfoItemeinä. Jokaisella potilaalla, havainnolla ja laitteella on tietyt perustiedot. Potilaaseen linkittyy mielivaltaisen määrä havaintoja, ja jokainen havainto on tehty jollain laitteella.

6.2 Kenttien tunnistaminen

Jotta FHIR-resursseja voisi automaattisesti generoida O-DF:n perusteella, muuntimen tulee kyetä tunnistamaan vastaavat kentät sekä O-DF:ssä että FHIR:ssä. FHIR-resurssit ovat hyvin määrämittäisiä, mutta O-DF ei tiedosformaattina vaadi tiettyä kenttänimeä tietylle tiedolle. Älylaitteet voivat olla eri valmistajilta ja käyttää eri nimiä samoille asioille: esimerkiksi ”syke” voi olla ”syke” yhdessä laitteessa, mutta ”pulssi” toisessa, tai englanniksi ”pulse” ja ”heart rate”. Näiden kaikkien tulisi vastata yhtä FHIR:n havaintoa.

Ratkaisuja kenttien automaattiseen tunnistamiseen on kolme. Ensimmäinen on koodata muuntimeen monta eri synonyymia, jotka tunnistuvat samaksi FHIR-kentäksi. Tämän ratkaisun etu on hallittavuus: muunnin on täysin terveydenhoitoa tarjoavan tahon hallinnassa, jolloin he pystyvät lisäämään synonyymejä tarvittaessa. Kehittyneemmässä järjestelmässä voidaan tarjota myös mahdollisuus käsin tehtävään kenttien tunnistamiseen. Kun ennen tuntematon laite yhdistetään järjestelmään, ihminen voi varmistaa, että automaattisesti tunnistetut kentät vastaavat oikeasti toisiaan ja yhdistää tuntemattomat kentät vastaaviin FHIR-kenttiin, jonka jälkeen järjestelmä osaa tulevaisuudessa yhdistää ne automaattisesti.

Ongelmaa voi lievittää laitteiden puolella. O-DF mahdollistaa monen nimen antamisen InfoItemeille, jolloin yksittäisen datakentän voi nimetä eri järjestelmien ja organisaatioiden käyttämillä nimillä. Esimerkki tästä on listauksessa 6.3, jossa valmistaja-InfoItemille on annettu monta nimeä. Valmistajat voivat ohjelmistopäivityksillä lisätä nimiä, jos tarvetta ilmenee. Kun laite tarjoaa tiedoille monta synonyymia, kasvaa todennäköisyys sille, että vastaanottava järjestelmä tunnistaa kentän automaattisesti. Haittapuolena jokaiseen lähtevään O-DF-tietoon lisätään turhia nimikenttiä. Yksittäisessä viestissä tästä ei ole paljon haittaa, mutta esineiden internetin mittakaavassa tavut kertautuvat, ja paljon kaistanleveyttä kuluu turhan tiedon välittämiseen. Tämä ongelma voidaan ratkaista lähettämällä oletuksena vain päänimi, ja lisänimet vain pyydettäessä.

```
<InfoItem name="manufacturer">
  <name lang="fi">Valmistaja</name>
  <name>Brand</name>
  <value type="xs:string">Acme Inc.</value>
</InfoItem>
```

Listaus 6.3: Monta nimeä yhdellä InfoItemillä.

O-DF:ssä ei ole määritelty nimielementeille mitään attribuutteja, mutta mielivaltaiset attribuutit on sallittu. Jos lisänimiä annetaan jonkin tietyn järjestelmän mukaan tai tietyllä kielellä, ne voidaan merkata attribuuteilla. Lisätauksessa 6.3 on tästä esimerkki. Yhdelle nimielementeistä on annettu attribuutti *lang* arvolla *fi*, mikä kertoo kentän arvon olevan suomeksi. Kuten aiemmin mainittiin, lisäattribuutteja ei määritellä standardissa, joten niiden tunnistamisessa on sama ongelma kuin muissakin kentissä.

Paras tapa tunnistaa kenttiä on nimien standardointi. Terveystieteenhuollossa käytetään Logical Observation Identifiers Names and Codes (LOINC) -tietokantaa¹, joka tarjoaa universaalin standardin lääketieteellisten havaintojen identifiointiin. Antamalla kaikille lääketieteellisille havainnoille oikea LOINC-koodi ne voidaan tunnistaa automaattisesti ja luotettavasti. Esimerkiksi LOINC-koodi 8867-4 tarkoittaa sykettä. Tällöin ongelmaksi jää tieto, jota LOINC ei kata. Esimerkiksi laitteiden metatieto, kuten valmistaja ja malli, ei kuulu LOINC:n piiriin. Nämä tiedot pitää standardoida jollain muulla menetelmällä. Jos O-DF:n käyttö yleistyy, hyvin todennäköisesti yleisten kenttien nimille syntyy epävirallinen standardi.

Koska tässä työssä toteutettiin sekä laitteet että muunnin, kenttien luotettava tunnistaminen ei ollut ongelma. Käytetty Aallon O-MI-referenssipalvelin ei tukenut mielivaltaisia attribuutteja nimielementeille eikä se tallentanut lähetettyjä ylimääräisiä nimielementtejä. Tämän vuoksi nimielementtejä lähetettiin vain yksi. Jos kentälle oli olemassa LOINC-koodi, sitä käytettiin nimenä; muissa tapauksissa käytettiin englanninkielistä nimeä.

Jos järjestelmä toteutetaan tuotantokäyttöön, kenttien tunnistamisongelma on todellinen, varsinkin jos pitää pystyä tukemaan mitä tahansa O-DF/O-MI-laitetta. Tällöin ainoa järjestelmää toimittavan tahon hallinnassa oleva keino on synonyymien yhdistäminen muuntimessa, mikä voi pahimmillaan vaatia käsityötä jokaisen uuden laitemallin kanssa. Kenttänimien standardointi on paras ratkaisu, mutta vaatii kaikkien toimijoiden yhteistyötä.

6.3 Laitteiden rekisteröinti

FHIR-havaintoresursseihin voidaan merkitä, millä laitteella ne on mitattu. Laitetta kuvaa FHIR:ssä laiteresurssi. Tämän työn järjestelmässä muunnin luo laitteille resurssit, kun ensimmäinen laitetta käyttävä mittausstulos vastaanotetaan. Jotta tämä on mahdollista, muuntimen pitää tietää, on-

¹<http://loinc.org/background>

ko laitteelle jo olemassa resurssia, ja tarvittaessa pystyä hakemaan O-MI-palvelimelta laitteen tiedot sen luomista varten.

Muunnin tekee tilauksia O-DF-mittauksiin. O-MI:n mukaisesti päivitysviesteihin sisällytetään vain muuttuneet kentät, ja laitetieto muuttuu harvoin. Muuntimen täytyy päivityksen saapuessa joko muistaa tai pyytää erikseen O-MI-palvelimelta laitteen tunniste. Kun se on saatu, muunnin kysyy FHIR-palvelimelta, onko laitteelle jo resurssia. Jos ei, muunnin lukee O-MI-palvelimelta laitteen olion ja luo sen perusteella uuden laiteresurssin. Listauksesta 6.2 nähdään, että viittaukset laitteisiin ovat InfoItmeitä, joiden arvona on URI, josta laitetta kuvaava O-DF-olio löytyy. Sille voidaan tehdä O-MI-lukupyyntö, jonka vastauksen perusteella laiteresurssi voidaan luoda.

6.4 Mittausten yksiköt

Hyvien FHIR-havaintojen tekemiseksi muuntimen pitää tietää, mitä yksikköä mittauksessa on käytetty. Joskus se on ilmeistä: askelia mitataan kappaleissa, sykettä lyönteinä minuuteissa. Joissain tapauksissa yksikkö riippuu maasta tai käyttäjän tottumuksesta: vaaka voi lähettää tuloksensa kilogrammoissa tai paunoissa, ja ranneke kuluneen energian kilokaloreissa tai jouleissa. O-DF:ssä mittausten yksiköt esitetään InfoItmin metadatatassa, jota ei lähetetä kuin pyydettyäessä. Täten muuntimen täytyy hakea metadata tarvittaessa tekemällä lukupyyntö.

Yksiköillä on sama tunnistamisongelma kuin muillakin kentillä. Unified Code for Units of Measure (UCUM) -koodijärjestelmällä² on mahdollista esittää yksiköt yksiselitteisesti sekä ihmisille että koneille, joten UCUM-koodeja on käytetty älylaitteiden metadatatassa.

6.5 FHIR-resurssien linkittäminen

Työssä käytettiin kolmea FHIR-resurssia: potilasta, havaintoa ja laitetta. Kun muunnin saa mittaustuloksen O-MI-palvelimelta, se tekee siitä havaintoresurssin ja linkittää sen potilas- ja laiteresursseihin. Jotta se voi tehdä tämän, sen pitää tietää, mikä O-DF-tunniste vastaa mitäkin tunnistetta FHIR:ssä.

²<http://unitsofmeasure.org/trac>

FHIR-resursseille on mahdollista antaa monta tunnistetta. Tallentamalla O-DF:ssä käytetty tunniste resurssiin, O-DF-oliota vastaava FHIR-resurssi voidaan löytää helposti. Listauksessa 6.4 on tästä esimerkki. Id-elementissä on palvelimen resurssille automaattisesti generoitu uniikki tunnus. Identifier-elementin sisällä on O-DF:ssä käytetty tunniste. Use-alielementti kertoo, missä kontekstissa tunnistetta voidaan käyttää. Arvo *secondary* tarkoittaa, että tunniste on toissijainen, ja siihen voi luottaa vain tietyissä konteksteissa: tässä tapauksessa O-DF-olioiden tunnistamisessa.

```
<Patient xmlns="http://hl7.org/fhir">
  <id value="30ec40af-e836-4597-9757-8c67639dae"/>
  <identifier>
    <use value="secondary"/>
    <value value="Paavo1"/>
  </identifier>
  <!-- Tiedot potilaasta -->
</Patient>
```

Listaus 6.4: Potilasresurssi, jossa O-DF-tunniste.

Kun resursseihin on lisätty O-DF:ssä käytetyt tunnistet, FHIR-palvelimelle voidaan tehdä hakuja niiden perusteella. Esimerkiksi voidaan hakea kaikki potilasresurssit, joilla on tietty tunniste. Kun haluttu resurssi on vastaanotettu, siitä saadaan FHIR:n sisäinen ID, jota voidaan käyttää resurssien linkittämiseen keskenään. Listauksessa 6.5 on sykehavainto, joka on linkitetty listauksen 6.4 potilaaseen ja lisäksi laitteeseen. Käytetyt linkit ovat suhteellisia polkuja resursseihin palvelimella.

Tässä työssä siirretyn datan määrä oli pieni, joten minkäänlaista tietokantaa tai välimuistia ei toteutettu muuntimeen. Kun uusi päivitys tulee, muunnin pyytää aina O-MI-palvelimelta siihen liittyvän potilaan ja laitteen. Tämän jälkeen se hakee FHIR-palvelimelta O-DF-tunnisteita vastaavat FHIR-resurssit, ja käyttää niiden FHIR-tunnisteita linkkien luomiseen.

```
<Observation xmlns="http://hl7.org/fhir">
  <id value="19847d9e-ebcc-4812-9bf8-495d2258a0"/>
  <code>
    <coding>
      <system value="http://loinc.org"/>
      <code value="8462-4"/>
    </coding>
  </code>
  <subject>
    <reference value="Patient/30ec40af-e836
      -4597-9757-8c67639dae"/>
  </subject>
  <device>
    <reference value="Device/29ef3b03-737b-43f5
      -8935-7c2e5a4674"/>
  </device>
  <!-- Mittaustulos -->
</Observation>
```

Lista 6.5: Sykehavainto, joka on linkitetty laitteeseen ja potilaaseen.

Luku 7

Pohdinta

Tässä luvussa pohditaan saatuja tuloksia. Luku 7.1 pohtii, voiko kokeiltua O-DF–FHIR-muuntoa käyttää tuotantoympäristössä. Luvussa 7.2 esitetään mahdollisia jatkokehitysideoita järjestelmälle.

7.1 Sovellettavuus

Järjestelmän pääsovelluskohde on ”tuo oma laitteesi” -terveydenhuolto, jossa hyödynnetään käyttäjällä jo olevia laitteita. Jotta järjestelmä kannattaa toteuttaa tuotantokäyttöön, O-MI:a luontaisesti käyttäviä laitteita pitäisi olla markkinoilla. Tämän työn kirjoitushetkellä niitä ei ole, eikä tiedossa ei ole yhtään laitevalmistajaa, jolla olisi suunnitelmia sisällyttää se laitteisiinsa. Laitevalmistajat haluavat usein tarjota ekosysteemejä, missä kaikki laitteet ovat joko heidän valmistamiaan, tai joihin laitteiden valmistamiseksi ulkopuolisten pitää maksaa lisensointimaksuja. Avointen standardien käyttöönotto ei välttämättä edistä heidän liiketoiminnallisia tavoitteitaan. Toisaalta jos jokin avoin standardi saavuttaa kriittisen massan, kaikkien on tuettava sitä, jotta heidän tuotteensa olisivat kilpailukykyisiä.

Mille tahansa laitteelle, jolla on sopivat ohjelmointirajapinnat, voidaan toteuttaa sovellus, joka tarjoaa datan O-DF/O-MI-muodossa. Jos tässä työssä kuvattu järjestelmä halutaan toteuttaa, näille laitteille voidaan tehdä ohjelmat järjestelmän kanssa kommunikoimiseksi. Toisaalta jos laitekohtaisia ohjelmistoja joudutaan tekemään, niin silloin ne voi noin samalla määrällä työtä räätälöidä toimimaan suoraan terveydenhuollon tietojärjestelmän kanssa (esim. kommunikoimalla FHIR:llä), joten O-DF/O-MI:n toteuttaminen ei ole hyödyllistä.

Kevyempi vaihtoehto on käyttää erillistä O-MI-palvelinta, jolle laitteet ohjelmoidaan kirjoittamaan päivityksensä. Tässä tapauksessa laitteiden ei tarvitse toteuttaa O-MI:sta muuta kuin kirjoitusviesti, eikä välttämättä edes sitäkään, jos palvelin tukee muitakin tapoja kirjoittaa tietoa. Tämä vaihtoehto voidaan myös toteuttaa pilvessä: esimerkiksi laitevalmistaja voi yhdistää kaikki myymänsä laitteet omaan pilveensä, ja tarjota pilvessä kaikelle datalle O-DF/O-MI-rajapinnan.

On mahdollista, että jokin yritys alkaa tarjota väliohjelmistoja, jotka muuntavat yleisten laitteiden datat O-DF-muotoon ja lähettävät ne O-MI-palvelimelle, josta ulkoiset järjestelmät voivat käyttää niitä. Etuna tällaisessa ratkaisussa on, että väliohjelmisto ei ole sidottu mihinkään tietyn toimialan kontekstiin. Esimerkiksi samaa väliohjelmistoa voidaan käyttää kotiautomaation tai teollisuuden laitteiden datan hallintaan, mikä laajentaa sitä tarjoavan yrityksen asiakaspohjaa. Yrityksen asiakkaat maksaisivat siitä, että he voivat muuntaa monien eri lähteiden datan yhteen ymmärrettävään muotoon jatkokäsittelyä varten. Väliohjelmistoja tarjoava yritys toimisi myös eräänlaisena epävirallisena standardointiorganisaationa, sillä se voi muuntaa kaikkien tukemiensa laitteiden datan käyttämään samoja O-DF-kenttänimiä. Lisäksi yritys voi myydä yksinkertaisia O-MI-palvelimia, joiden käyttö ei vaadi teknistä osaamista. Loppukäyttäjät voivat käyttää niitä toimimaan kodin tai toimiston datan keskuksina.

Yksi vaihtoehto on tehdä muunnin siten, että se ymmärtää eri laitteiden eri tapoja tarjota tietoa. Tällainen muunnin on huomattavasti vaikeampi toteuttaa, koska joudutaan toteuttamaan tuki monelle tavalle kommunikoida, eikä ole esimerkiksi standardoitua tapaa tilata laitteilta päivityksiä. Data ei myöskään välttämättä ole samassa muodossa.

7.2 Jatkokehitys

Kaikki laitteet olivat simuloituja tässä työssä. Järjestelmän muunnin voi periaatteessa toimia oikeiden laitteiden kanssa, mutta sitä ei ole kokeiltu. Järjestelmän käyttäminen fyysisten laitteiden kanssa on looginen seuraava askel. Paras tapa tehdä se on kirjoittaa esimerkiksi jollekin älykellolle O-DF/O-MI-ohjelma, joka tekee laitteesta O-MI-noodin ja käyttää sitä järjestelmän kanssa.

Luvussa 5.4 esitetyt rajaukset yksinkertaistavat järjestelmää paljon, mutta ne poistavat myös tarpeellisia ominaisuuksia. Esimerkiksi tietoturva on terveystiedon käsittelyssä hyvin tärkeää, mutta siihen ei otettu lainkaan kan-

taa. Jatkokehityksessä järjestelmän pitää pystyä pitämään tieto asiattomien ulottomattomissa.

Tässä työssä kaikki data on kulkenut älylaitteilta tietojärjestelmään, mutta tieto voi kulkea toiseenkin suuntaan. Kaikki simuloidut laitteet olivat sensoreita, jotka vain havaitsivat ympäristöään, mutta jo nyt on olemassa laitteita, jotka voivat vaikuttaa ympäristöönsä. Jos näitä laitteita halutaan ohjata etänä, lääkärin käyttämän järjestelmän pitää pystyä lähettämään dataa laitteille.

Luku 8

Päätäntö

Tässä työssä tutkittiin älykkäiden terveyslaitteiden yhteensovittamista terveydenhuollon tietojärjestelmien kanssa. Älylaitteiden standardiksi valittiin O-MI/O-DF ja tietojärjestelmälle FHIR. Tutkimusta varten toteutettiin ohjelma, joka ottaa vastaan O-DF-muotoista dataa O-MI:n yli, muuntaa sen FHIR-muotoon ja lähettää sen eteenpäin FHIR-palvelimelle.

Teknisessä toteutuksessa suurimmat haasteet olivat O-DF-datan esitysmuodon valinta ja kenttien vastaavuuden tunnistaminen O-DF:ssä ja FHIR:ssä. O-DF-datan parhaaksi esitysmuodoksi valikoitui potilaslähtöinen esitysmuoto, jossa data esitetään potilaaseen liittyvänä. Datan saatavuuden parantamiseksi yhteen potilaaseen liittyvä data kannattaa myös keskittää yhdelle O-MI-solmulle. Terveysdataan liittyvien kenttien tunnistamisessa on parasta käyttää lääketieteen LOINC-koodeja. Muille kentille on mahdollista esittää synonyymejä O-DF:ssä tai koodata vaihtoehtoja O-DF–FHIR-muuntoon. O-DF:ssä yleisien kenttien nimille olisi kehitettävä standardi tai käytäntö, jotta automaattinen lukeminen olisi mahdollisimman sujuvaa.

Kehitetyn järjestelmän sovellettavuus tuotannossa on tällä hetkellä heikko. O-MI/O-DF:ää vakiona käytettäviä laitteita ei ole. Monille laitteille on mahdollista tehdä sovellus, joka tarjoaa datan oikeassa muodossa tai lähettää sen eteenpäin O-MI-solmulle. Sellaisen toteuttaminen ei kuitenkaan ole merkittävästi hyödyllisempää kuin suoraan tietojärjestelmän kanssa kommunikoivan sovelluksen toteuttaminen.

Tutkimus toteutettiin simuloimalla älylaitteita. On todennäköistä, että simulointi yksinkertaisti toteutusta liikaa, joten jatkotutkimuksena järjestelmää tulisi kokeilla oikeilla laitteilla. Lisäksi montaa tärkeää osa-aluetta, kuten tietoturva, ei toteutettu ollenkaan, joten ne pitäisi myös sisällyttää järjestelmään.

Esineiden internet kasvaa nopeasti, ja sitä tullaan soveltamaan lähes kaikilla elämäntiloilla. On siten tärkeää, että esineiden älykkyyttä voidaan hyödyntää mahdollisimman helposti. Tämän työn tarkoitus on ollut selvittää, miten se voidaan tehdä terveydenhuollossa. Työtä on tehtävä vielä paljon, ennen kuin esineiden internetin täydet hyödyt voidaan ottaa käyttöön.

Kirjallisuutta

- [1] ALANI, M. M. *Guide to OSI and TCP/IP Models*. Springer International Publishing, 2014.
- [2] AMERICAN REGISTRY FOR INTERNET NUMBERS. Arin ipv4 free pool reaches zero. <https://www.arin.net/announcements/2015/20150924.html>. Viitattu 13.5.2016.
- [3] APOTTI-HANKE. Mikä on apotti? <http://www.apotti.fi/apotti-hanke/>. Viitattu 15.7.2016.
- [4] BRADLEY, J., BARBIER, J., AND HANDLER, D. White paper: Embracing the internet of everything to capture your share of \$14.4 trillion. Tech. rep., Cisco, 2013. http://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoE_Economy.pdf Viitattu 13.5.2016.
- [5] CORNO, F., RUSSIS, L. D., AND ROFFARELLO, A. M. A healthcare support system for assisted living facilities: An iot solution. In *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)* (June 2016), vol. 1, pp. 344–352.
- [6] DARBY, S. The effectiveness of feedback on energy consumption: a review for defra of the literature on metering, billing and direct displays. Tech. rep., Environmental Change Institute, University of Oxford, huhtikuu 2006.
- [7] DAS, S., BALLAV, M., AND KARFA, S. Application of iot in detecting health risks due to flickering artificial lights. In *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on* (Aug 2015), pp. 2331–2334.
- [8] DAVE, B., KUBLER, S., FRÄMLING, K., AND KOSKELA, L. Opportunities for enhanced lean construction management using internet of things standards. *Automation in Construction* 61 (2016), 86 – 97.

- [9] FARRA, R., SHEPPARD, N. F., MCCABE, L., NEER, R. M., ANDERSON, J. M., SANTINI, J. T., CIMA, M. J., AND LANGER, R. First-in-human testing of a wirelessly controlled drug delivery microchip. *Science Translational Medicine* 4, 122 (2012), 122ra21–122ra21.
- [10] FILIPPOV, R. Security model for the open messaging interface (o-mi) protocol. Master's thesis, Aalto-yliopisto, 2016.
- [11] GAMMA, E., HELM, R., JOHNSON, R., AND VISSIDES, J. *Olio-ohjelmointi: suunnittelumallit*. IT Press, 2001.
- [12] GARTNER. Gartner says 6.4 billion connected things will be in use in 2016, up 30 percent from 2015. <http://www.gartner.com/newsroom/id/3165317>, marraskuu 2015. Viitattu 13.5.2016.
- [13] HEALTH LEVEL SEVEN INTERNATIONAL. Base resource definitions. <https://www.hl7.org/fhir/resource.html>, lokakuu 2015. Viitattu 4.7.2016.
- [14] HEALTH LEVEL SEVEN INTERNATIONAL. Fhir overview. <https://www.hl7.org/fhir/overview.html>, lokakuu 2015. Viitattu 4.7.2016.
- [15] HEALTH LEVEL SEVEN INTERNATIONAL. Fhir timelines. <https://www.hl7.org/fhir/timelines.html>, lokakuu 2015. Viitattu 30.6.2016.
- [16] HEALTH LEVEL SEVEN INTERNATIONAL. <http://www.hl7.org/implement/standards/fhir/summary.html>. <http://www.hl7.org/implement/standards/fhir/summary.html>, lokakuu 2015. Viitattu 4.7.2016.
- [17] HEALTH LEVEL SEVEN INTERNATIONAL. Resource device - content. <https://www.hl7.org/fhir/device.html>, lokakuu 2015. Viitattu 4.9.2016.
- [18] HEALTH LEVEL SEVEN INTERNATIONAL. Resource formats. <https://www.hl7.org/fhir/formats.html>, lokakuu 2015. Viitattu 4.7.2016.
- [19] HEALTH LEVEL SEVEN INTERNATIONAL. Resource observation - content. <https://www.hl7.org/fhir/observation.html>, lokakuu 2015. Viitattu 4.9.2016.
- [20] HEALTH LEVEL SEVEN INTERNATIONAL. Resource patient - content. <https://www.hl7.org/fhir/patient.html>, lokakuu 2015. Viitattu 16.7.2016.

- [21] HEALTH LEVEL SEVEN INTERNATIONAL. Restful api. <https://www.hl7.org/fhir/http.html>, lokakuu 2015. Viitattu 12.7.2016.
- [22] HEALTH LEVEL SEVEN INTERNATIONAL. Search. <https://www.hl7.org/fhir/search.html>, lokakuu 2015. Viitattu 2.9.2016.
- [23] HEALTH LEVEL SEVEN INTERNATIONAL. Fhir version history. <https://www.hl7.org/fhir/history.html>, toukokuu 2016. Viitattu 30.6.2016.
- [24] HEALTH LEVEL SEVEN INTERNATIONAL. Hl7 version 3 product suite. https://www.hl7.org/implement/standards/product_brief.cfm?product_id=186, 2016. Viitattu 4.7.2016.
- [25] INFOR. A real-world fhir implementation. <http://www.infor.com/articles/infor-hackensackumc-build-real-world-fhir-implementation>, heinäkuu 2015. Viitattu 4.7.2016.
- [26] JUNIPER RESEARCH. Internet of things connected devices to almost triple to over 38 billion units by 2020. <http://www.juniperresearch.com/press/press-releases/iot-connected-devices-to-triple-to-38-bn-by-2020>, July 2015. Viitattu 13.5.2016.
- [27] KANSAINVÄLINEN TELEVIESTINTÄLIITTO. Key 2005–2015 data for the world, by geographic regions and by level of development. <http://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx>, 2015. Viitattu 13.5.2016.
- [28] KOPETZ, H. *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Springer US, 2007.
- [29] KUBLER, S., FRÄMLING, K., AND DERIGENT, W. P2p data synchronization for product lifecycle management. *Computers in Industry* 66 (2015), 82–98.
- [30] KUMAR, K. M. C. Internet of fitness things – a move towards quantified health: Concept, sensor-cloud network, protocols and a new methodology for osa patients. In *2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)* (Dec 2015), pp. 364–369.
- [31] LAMPRINAKOS, G. C., MOUSAS, A. S., KAPSALIS, A. P., KAKLAMANI, D. I., VENIERIS, I. S., BOUFIS, A. D., KARMIRIS, P. D., AND MANTZOURATOS, S. G. Using fhir to develop a healthcare mobile

- application. In *Wireless Mobile Communication and Healthcare (Mobi-health)*, 2014 EAI 4th International Conference on (Nov 2014), pp. 132–135.
- [32] MOSER, K., HARDER, J., AND KOO, S. G. M. Internet of things in home automation and energy efficient smart home technologies. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (Oct 2014), pp. 1260–1265.
- [33] RAY, P. P. Home health hub internet of things (h3iot): An architectural framework for monitoring health of elderly people. In *Science Engineering and Management Research (ICSEMR)*, 2014 International Conference on (Nov 2014), pp. 1–3.
- [34] RUMINSKI, J., BUJNOWSKI, A., KOCEJKO, T., ANDRUSHEVICH, A., BIALLAS, M., AND KISTLER, R. The data exchange between smart glasses and healthcare information systems using the hl7 fhir standard. In *2016 9th International Conference on Human System Interactions (HSI)* (July 2016), pp. 525–531.
- [35] SZYMANSKI, T. H. Supporting consumer services in a deterministic industrial internet core network. *IEEE Communications Magazine* 54, 6 (June 2016), 110–117.
- [36] THE OPEN GROUP. Open data format (o-df) standard, 2014.
- [37] THE OPEN GROUP. Open messaging interface (o-mi), an open group internet of things (iot) standard, 2014.
- [38] ULLAH, K., SHAH, M. A., AND ZHANG, S. Effective ways to use internet of things in the field of medical and smart health care. In *2016 International Conference on Intelligent Systems Engineering (ICISE)* (Jan 2016), pp. 372–379.
- [39] VERMESAN, O., FRIESS, P., GUILLEMIN, P., GUSMEROLI, S., SUNDMAEKER, H., BASSI, A., JUBERT, S. I., MAZURA, M., HARRISON, M., EISENHAUER, M., AND DOODY, P. Internet of things strategic research roadmap. Tech. rep., European Research Cluster on the Internet of Things, 2011. http://www.internet-of-things-research.eu/pdf/IoT_Cluster_Strategic_Research_Agenda_2011.pdf. Viitattu 14.5.2016.